Engineering Science and Technology, an International Journal 24 (2021) 848–859

HOSTED BY

ELSEVIER

Contents lists available at ScienceDirect

Engineering Science and Technology, an International Journal

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

Full Length Article

Market sentiment-aware deep reinforcement learning approach for stock portfolio allocation

Prahlad Koratamaddi [a,*], Karan Wadhwani [a], Mridul Gupta [a], Sriram G. Sanjeevi [b]

[a] National Institute of Technology, Warangal, India
[b] Department of Computer Science and Engineering, National Institute of Technology, Warangal, India

ARTICLE INFO

Article history:
Received 4 August 2020
Revised 30 December 2020
Accepted 10 January 2021
Available online 26 March 2021

Keywords:
Reinforcement learning
Portfolio allocation
Sentiment analysis
Deep learning
Stock trading

ABSTRACT

The stock market currently remains one of the most difficult systems to model in finance. Hence, it is a challenge to solve stock portfolio allocation wherein an optimal investment strategy must be found for a curated collection of stocks that effectively maximizes return while minimizing the risk involved. Deep reinforcement learning approaches have shown promising results when used to automate portfolio allocation, by training an intelligent agent on historical stock prices. However, modern investors are actively engaging with digital platforms such as social media and online news websites to understand and better analyze portfolios. The overall attitude thus formed by investors toward a particular stock or financial market is known as market sentiment. Existing approaches do not incorporate market sentiment which has been empirically shown to influence investor decisions. In our paper, we propose a novel deep reinforcement learning approach to effectively train an intelligent automated trader, that not only uses the historical stock price data but also perceives market sentiment for a stock portfolio consisting of the Dow Jones companies. We demonstrate that our approach is more robust in comparison to existing baselines across standardized metrics such as the Sharpe ratio and annualized investment return.

© 2021 Karabuk University. Publishing services by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

1. Introduction

1.1. Motivation

Portfolio allocation is one of the most challenging and interesting problems in modern finance. This is because the stock market is a complex system [1], permeated by a web of interrelated return effects that require substantial amounts of computational effort to disentangle and model return regularities. Additionally, the stock market keeps evolving constantly – therefore, predicting stock price movements is not a trivial task. To achieve an optimal solution to portfolio allocation, a trader must constantly be able to diversify and reallocate funds among the stocks in their portfolio to maximize profit, while simultaneously minimizing risk [2]. Finding such a profitable yet low-risk trading strategy is certainly one of the best ways to ensure financial growth. Hence, several investment management companies are continuously attempting to solve this problem in increasingly better ways, using more sophisticated approaches.

Some of the earliest works on portfolio allocation were based on mathematical models [4] that leveraged techniques such as quadratic programming, stochastic calculus, numerical analysis, etc. Initially, statistical learning methods were used for simple solutions that involved numerical analysis [9] – for instance, the Newton-Raphson algorithm was used to solve logistic regression. However, in the 1990s, with the popularization of supervised machine learning tools such as artificial neural networks, several deep learning approaches were proposed for various applications on the stock market [10–12]. The success of neural networks compared to other machine learning approaches in predicting stock returns is attributed to their ability to learn complex non-linear functions [13].

The general approach followed by studies that used supervised learning to solve portfolio allocation, is based on trading via forecasting. This approach consists of two steps -.

1. Forecasting: develop a predictive model by using historical data on the assets' prices and other relevant features for training and forecast the assets' price change after a particular period.
2. Decision making: use the prediction of the assets' price to take a decision on the trading action – buy, hold, or sell.

* Corresponding author.
E-mail addresses: kprahlad@student.nitw.ac.in (P. Koratamaddi), sgs@nitw.ac.in (S.G. Sanjeevi).

https://doi.org/10.1016/j.jestch.2021.01.007
2215-0986/© 2021 Karabuk University. Publishing services by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

This simple and popular two-step approach may seem ideal at first, but it may lead to sub-optimal performance [3] due to certain key limitations. Minimizing the forecasting error (as part of the first step) is not the final objective of a system that aims to solve portfolio allocation. Rather, the decision making step becomes a more crucial component of the entire process. The common practise followed here that only forecasts obtained from the first step would be used for taking decisions leads to what is referred to as a "forecast bottleneck" [3], due to a loss of information available to the decision making module, relative to the forecasting module.

In the recent years, reinforcement learning has emerged to be a powerful tool to solve problems that involve sequential decision making, and has been used to develop intelligent agents that could learn complex strategies [14]. Portfolio allocation can be modelled as a Markov Decision Process (MDP) in reinforcement learning – this approach not only combines the two steps required for the supervised learning approaches into a single integrated step, much in line with the thinking of a real investor, but also overcomes the limitations of the traditional supervised learning approaches. Basic reinforcement learning, however, struggles to scale with the large extent of information processing required for most real-life problems. Recently, the adoption of deep learning techniques is taking place along with reinforcement learning in order to help it scale to previously intractable problems. This is done in two main ways – advanced feature extraction and function approximation [14]. For our work, we follow a deep reinforcement learning approach that is based on the latter.

Over the years, purely quantitative models have evolved to improve in performance, but have always considered only a part of the picture – they assume that an investor takes decisions solely based on quantitative inputs like stock prices, company performance metrics, and other technical factors. In reality, investors are also shown to be influenced by many qualitative factors [31] such as the general outlook of the market, company reputation and brand, shareholder satisfaction and so on. These factors are encapsulated in a term called "market sentiment" or "investor sentiment" [15], which is a measure of the collective opinion of investors on a stock or financial market. With the advent of the internet and social media, communication and information sharing have enabled investors to effectively scrutinize and analyze their financial assets in detail to take better, more informed decisions. The resulting availability of massive amounts of unstructured data allows models to effectively gauge market sentiment, providing a more accurate simulation of the actual market dynamics at play.

In this paper, we propose an approach wherein our trader receives an external market stimulus, similar to how a real-life trader perceives it from available sources such as conventional news media, and social media. We consider a portfolio of 30 companies from the Dow Jones Industrial Average (DJIA) index based on the U. S. stock market which are known to be some of the most influential and reputed. We propose a sentiment-aware deep reinforcement learning approach that builds upon the adaptive deep deterministic policy gradients (DDPG) algorithm [48] and learns to dynamically utilize perceived market sentiment from the real world. We aim to analyze the effect of incorporating market sentiment, a qualitative factor, into a quantitative deep reinforcement learning model.

1.2. Outline

The remainder of this paper is organized as follows – in Section 2, we cover the necessary background details on portfolio allocation, describe our problem statement, discuss relevant research on market sentiment analysis and introduce deep reinforcement learning approaches. We further explain the specifics of our proposed approach in Section 3, and in Section 4 we detail the techniques used to acquire and process the data obtained. We discuss the results obtained in Section 5 and conclude in Section 6 by highlighting our key contributions, summarizing our work, and providing the scope for future research.

2. Background

2.1. Stock Portfolio Allocation

Stock trading is one of the most common ways that investors endeavor to achieve financial growth. The collection of stocks that an investor chooses to track, analyze, allocate and redistribute funds is known as a stock portfolio. In general, because of the volatility and uncertainty associated with price movements of the stock market, a certain amount of risk is expected with investments on a stock portfolio, i.e., along with the potential for great returns, there is a chance that the value of the investment may decline or not perform as expected. Thus, ideally, an investor would aim to find an investment strategy that aims to maximize returns by taking the least possible risk. The implementation of such a strategy, that considers the time horizon, risk tolerance and investment goals is referred to as portfolio allocation [20].

The pioneering work of Harry Markovitz introduced the modern portfolio theory (MPT) [2], an optimization framework for obtaining efficiently diversified portfolios. Other notable developments include – the Sharpe diagonal model [5] that simplified the computation involved in the Markowitz model using a single market index, a multi-index model by Cohen and Pogue [6], and extensions to the diagonal model [7,8], among others.

Markovitz's modern portfolio theory (MPT) is extensively used by investors to this day due to its many practical benefits – it helps build a diversified portfolio, and opt for a more effective portfolio using a plot known as the efficient frontier [2]. MPT laid the groundwork for all research involving quantitative analysis on portfolio allocation. It provides a means to construct a portfolio of assets according to the goals of an investor – to maximize returns at a given level of risk, or to minimize risk at a given level of expected return. One of the many important conclusions to Markovitz's work is the idea that diversification of assets improves allocation outcomes. This is arrived at based on statistical measures such as variance and correlation, that reveal the fact that the influence of an investment on the entire portfolio is more significant than its individual performance. In other words, this demonstrates that it's the contribution of the risk of various securities in the portfolio that matters to the risk-averse investor [23].

We now proceed to discuss the two main variants of the MPT, that would also serve as two of our benchmark portfolio allocation strategies.

2.1.1. Mean variance analysis

Mean variance analysis provides a mathematical framework to the investor to weigh in risk, expressed as variance, against expected return [2]. The aim of mean variance analysis is to enable the investor to differentiate among several portfolios and evaluate investment decisions by calculating the expected return and portfolio variance.

Mathematical definitions offered by the model:

- Portfolio return is the proportion-weighted combination of the constituent assets' returns
- Portfolio volatility is a function of the correlations ρ_{ij} of the component assets, for all asset pairs (i, j).
- Expected return:

$$E(R_p) = \sum_i w_i E(R_i) \tag{1}$$

where R_p is the return on the portfolio, R_i is the return on asset i and w_i is the weighting of component asset i (that is, the proportion of asset "i" in the portfolio). Constraints for these weights is given as follows:

$$w_i \in [0, 0.2], i = 0, 1, 2, \ldots .m, \sum_{i=1}^{m} w_i = 1 \tag{2}$$

- Portfolio return variance:

$$\sigma_p^2 = \sum_i w_i^2 \sigma_i^2 + \sum_i \sum_{j \neq i} w_i w_j \sigma_i \sigma_j \rho_{ij}, \tag{3}$$

where σ is the (sample) standard deviation of the periodic returns on an asset, and ρ_{ij} is the correlation coefficient between the returns on assets i and j.

Sharpe ratio is one of the most common metrics that is used to weigh in investments. It represents a amount of return (in units) that can be obtained by taking a unit of risk. The objective of mean variance analysis, with this metric in consideration, is to find the portfolio allocation that maximizes the Sharpe ratio [24].

2.1.2. Min variance analysis

Min variance analysis is similar to mean variance analysis – the critical change being that the objective here is modified to solely face the least risk or in this case, portfolio return variance.

The above traditional methods thus reinforce the concept that a portfolio's risk profile is based on the contributions of individual securities to the entire portfolio.

2.1.3. Issues with modern portfolio theory

MPT was certainly one of the most influential and widely studied works in modern finance that offered significant benefits to investors. But, it is still a purely quantitative approach that assumes that the market is ideal and that investors do not make irrational choices [2].

Let us now look at three major limitations with MPT:

- In the real world, investors may not always make rational choices – for example, if a company's stock is trending and the majority of investors buy or sell that stock then the remaining investors in that social circle are also inclined to buy or sell that stock, which is what is known as "herd mentality" [25] as explained from the perspective of behavioural finance.
- The stock market is not "fully valid and efficient", contrary to the efficient market hypothesis (as assumed by MPT) and is practically bound to be fallible and inefficient [26]. The 2008 financial crisis is an example that illustrates a stock market crash. The market gets drastically affected due to depressions and recessions in the economy. There is a complex interplay of economic and social factors that affect the stock market, a key factor being the investor mood [27]. MPT does not consider such qualitative factors.
- Another important issue is the assumption of independence of individual securities in the portfolio. Many stocks have underlying connections and dependencies that influence market prices. This information, obtained from the market environment, is not considered in the solution offered by the Markowitz theory.

Thus, we observe that the inclusion of market sentiment is vital to overcome the above limitations and improve effectiveness in portfolio allocation.

2.2. Problem description

The problem in consideration is to find an effective portfolio allocation strategy, given the environmental inputs of the stock market – including historical stock prices, sentiment information and current portfolio characteristics such as the amount of holdings of each of the stock in the portfolio, and the available balance that the investor has. In terms of reinforcement learning terminology, the goal of our intelligent trader agent would be to develop an optimal policy that maximizes returns with minimal risk involved.

Our problem setting is as follows – the set of stocks we consider for the portfolio of our trader agent consists of 30 companies that have been a part of the Dow Jones Industrial Average index. These companies form a versatile portfolio that is representative of multiple industries – therefore, a successful performance of the trader agent with this portfolio could indicate its robustness across a wide range of companies. The timeline we consider is of the last two decades – from the year 2000 to the year 2019. We provide our trader agent with an initial portfolio balance of 10,000 U.S. dollars to facilitate the agent to begin trading.

The trader can buy, sell or hold stock related to any of the companies. The trader can take this action once in a day. We place a safe upper limit of 5 stocks the trader can buy, or sell at once, to limit extreme decisions from being taken by the agent. Fig. 1 illustrates the described stock market dynamics – the possible portfolios that can arise from the current portfolio. To capture the effect of the trader observing the real stock market, we provide environmental inputs that give suitable cues for both quantitative and qualitative aspects – specifically, for every eligible trading day of the timeline considered, the trader is given the measure of all closing prices of the Dow Jones stocks, and the market sentiments of the 30 companies derived from relevant Google News headlines and Twitter tweets.

In the described problem setting, the trader has to dynamically keep track of the portfolio and update it daily, taking the right

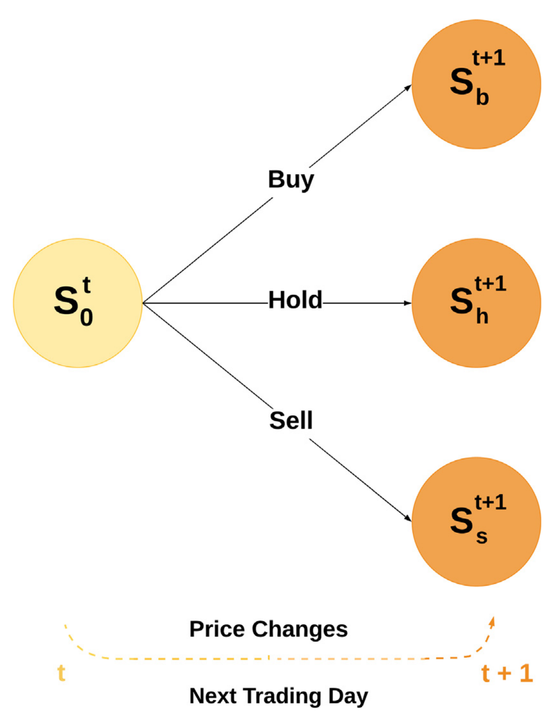

Fig. 1. S_0^t denotes the state of the portfolio as on day t, and $S_b^{t+1}, S_h^{t+1}, S_s^{t+1}$ are possible portfolio outcome states based on the actions buy, hold and sell taken by the trader on day t respectively.

investment decisions by fully learning to utilize the provided observations on the stock market, including market sentiments.

2.3. Sentiment analysis

The role of market sentiment in portfolio allocation has been a growing research topic. We discuss the relevant theory to highlight the influence of sentiment on traders when making investment-related decisions in the stock market.

Traditional portfolio allocation studies consider investors as being "unemotional" in their decisions, and that stock prices are affected by new information randomly [28]. However, recent empirical studies [29] suggest that stock prices do not fully follow random walks. Modern behavioural finance scientists believe that investors are thus practically "emotional" to some degree, and their cognitive biases fueled by external market sentiments drive the fluctuations in the perceived value of the company, which in turn influence the market price of the company's stock [30–32].

Market sentiment information is obtained from various sources such as conventional news and social media [21]. In the present age of information, social media platforms enable real-time updates and unprecedented speeds of communication seamlessly – this allows for individuals with a common interest to form communities. Micro-blogging sites like Twitter provide the platform for the investor community to share their opinions on companies. Eventually, some of these opinions become influential enough to drive stock price movements in the market. These drivers form a crucial part of the collective market sentiment that is of our particular interest. Similarly, as many investors also read through financial news articles and op-eds online, Google News offers many such drivers too. Therefore, a huge amount of opinionated investor sentiment data is present in the form of unstructured natural language text [33] in both of these sources – extracting and utilizing the collective sentiment from this abundant online data has given rise to two emerging fields – affective computing and sentiment analysis [34].

Studies such as Smailovic et al. [35] and Bollen et al. [37] have shown how performing sentiment analysis on a large Twitter dataset can be used to effectively predict stock market movements; Frank and Antweiler [36] have shown that the extracted sentiments on Yahoo! Finance postings have statistical significance in stock price prediction. Li et al. [38,39] have constructed a quantitative trader that uses publicly available web news and social media data, along with company-specific news sentiment data for forecasting stock price movements. Picasso et al. [40] have combined indicators of technical analysis and sentiment of news articles to build a robust predictive model. As can be seen, these studies leverage market sentiment to specifically address the problem of stock price forecasting.

Stock price forecasting only forms a part of portfolio allocation – taking the optimal investment decisions is the other major challenge, and, even as the relevance of market sentiment is continually increasing, relatively minimal work has been published on the problem of portfolio allocation that leverages market sentiment. We discuss some of these state-of-the-art works that were published in the last few years: Xing et al. [41] used Bayesian asset allocation on 5 companies by applying the Black Litterman Model. They also used an ECM-LSTM model for learning the expected returns and sentic computing's conceptual, sentence level sentiment analysis for asset allocation. Koyano and Ikeda [42] have proposed a new type of portfolio strategy which uses semi-supervised learning on posts in stock microblogs to maximize the cumulative returns using a Follow-the-Loser approach. Another recent work by Malandri et al. [43] compared multiple machine learning algorithms for portfolio allocation with 15 companies from New York Stock Exchange (NYSE) and sentiment data from StockFluence API. Therefore, several empirical research works support our intuition that investor sentiments do influence the market and can improve portfolio allocation performance. Furthermore, Cambria et al. [44] have recently proposed a state-of-the-art polarity detection approach called SenticNet 6, designed to not only utilize traditional bottom-up learning methods (known as subsymbolic AI tools) for predicting sequences of letters and words using deep learning, but also to integrate a top-down learning method (via symbolic logic) that introduces the model to logical reasoning, derived from a basic understanding of the world and social norms, cultural awareness, common sense knowledge, etc. The robustness of SenticNet 6 suggests that text-based polarity detection is becoming better at perceiving market sentiment, indicating a great potential for its usage to improve many financial applications, including portfolio allocation.

2.4. Deep reinforcement learning

Reinforcement learning is a machine learning paradigm wherein a software agent placed in an environment attempts at learning to maximize a notion of cumulative reward by a trial and error approach [45]. The agent observes its state in the environment, and takes an action based on its policy to move to the next state. The environment then gives it an immediate reward that causes the agent to update its strategy.

Deep reinforcement learning is a recent, rapidly growing approach wherein neural networks are used as function approximators for traditional reinforcement learning algorithms such as Q-learning, to help scale up cases involving a large set of states and actions in the environment [46]. It has been demonstrated by empirical studies that deep reinforcement learning approaches can successfully be applied to optimize profitable stock trading and portfolio allocation. Xiong et al. [47] used the deep deterministic policy gradients algorithm to improve trading profits significantly compared to min variance analysis and the Dow Jones Industrial average. Li et al. [48] introduced the adaptive deep deterministic policy gradients algorithm that improves upon the DDPG algorithm by leveraging prediction errors that indicate whether the market behavior was bullish or bearish. Yu et al. [49] proposed a novel model-based deep reinforcement learning algorithm which works with both on-policy and off-policy RL algorithms. The proposed architecture consists of an infused prediction module (IPM), a generative adversarial data augmentation module (DAM), and a behavior cloning module (BCM).

Motivated by the above research, our approach is also one that is based on deep reinforcement learning. We proceed to describe the basic concepts that are relevant to our deep reinforcement learning approach.

2.4.1. Markov Decision Process (MDP)

It is observed that portfolio allocation is a sequential decision-making process, as the trader would need to make investment choices every day, one day after the other. Thus, the problem of portfolio allocation is modelled as a Markov Decision Process (MDP). A finite MDP (as considered here) is a four tuple.

$(S, A, P_a(s, s'), R_a(s, s'))$ [50] defined as below:

- S is a finite set of states,
- A is a finite set of actions (and A_s is the finite set of actions available from state s)
- $P_a(s, s') = \Pr(s_{t+1} = s' | s_t = s, a_t = a)$ is the state transition probability, which is the probability that an action a in state s at time t will lead to state s' in time $t + 1$,
- $r_a(s, s') = E(r_t + 1 | s_t = s, a_t = a, s_{t+1} = s')$ is the expected immediate reward received after performing action a, leading to a transition from state s to state s'.

We proceed to define more relevant terms:

- $\pi(s, a)$ is the agent's policy – which is a probabilistic map defined on $A \times S \rightarrow [0, 1]$, as

  $$\pi(s, a) := \Pr(a_t = a | s_t = s)$$

  π^* is a common representation for the "optimal" policy that results in maximum cumulative reward.
- R_t is the discounted cumulative reward, called "return" obtained by the agent at time t expressed as

  $$R_t = \sum_{i=t}^{T} \gamma^{i-t} r_{a_i}(s_i, s_{i+1})$$

  where $\gamma \in [0, 1]$ is known as the discount rate, usually set close to 0.99.
- $Q_\pi(s, a)$ is the state action value function that denotes what the value (in terms of return) is of taking action a at state s as recommended by the policy π.

Fig. 2 shows simple state transitions in a finite MDP. At state S_0, action a_0 gives an immediate reward of r_0 and leads to state S_1 and so on.

Reinforcement learning can be used to solve an MDP, that is maximize the $Q_\pi(s, a)$ by finding out the optimal policy π^*. The equation given below:

$$Q_{t+1}(s_t, a_t) =$$

$$Q_t(s_t, a_t) + \alpha(r_{t+1} + \gamma \max_a Q_t(s_{t+1}, a) - Q_t(s_t, a_t))$$

where α is the learning rate, and γ is the discount factor, is known as the Bellman equation. It is implemented through dynamic programming approaches such as value iteration to maximize the Q value.

But, traditional methods like dynamic programming are ineffective when scaled up to problems involving large amounts of data and complicated strategies [45]. Thus, we now proceed to discuss the deep reinforcement learning algorithms that address this issue.

2.4.2. Deep Deterministic Policy Gradients (DDPG)

Deep Deterministic Policy Gradients (DDPG) is a deep reinforcement learning algorithm that concurrently learns a Q-function (learnt by the critic network) and a policy (learnt by the actor network). The algorithm is a model-free, off-policy actor-critic algorithm using deep function approximators that can learn policies in high-dimensional, continuous action spaces [51]. DDPG is especially preferable compared to algorithms such as DQN because of its ability to scale up to difficult problems in the real world, such as portfolio allocation involving continuous and real-valued spaces [51].

2.4.3. Adaptive Deep Deterministic Policy Gradients (Adaptive DDPG)

As an improvement to the DDPG algorithm that allowed for learning the bullish and bearish aspects of the market actively, Li et al. [48] propose a modified Rescorla-Wagner model-based adaptive approach [52] which can learn differently from positive and negative environments and can calculate the reward obtained by choosing different actions (buy, hold and sell). The model adjusts the amplitude of change in the Q-value per epoch, using different learning rates depending upon whether the prediction error was positive or negative. The model can distinguish between "good" and "bad" environment feedback(referred to as $RW\pm$) based on prediction errors – the good feedback is an indication of a bullish environment, and the bad feedback is an indication of a bearish environment. Thus, the model eventually learns from experience to behave according to the nature of the market.

In this model, the incremental updates made to the Q-function using the Bellman equation at the critic in the DDPG algorithm are leveraged as:

$$Q_\pi(s_{t+1}, a_{t+1}) = Q_\pi(s_t, a_t) + \alpha\delta(t)$$

and,

$$\delta(t) = r_(s_t, a_t, s_{t+1}) - Q_\pi(s_t, a_t)$$

where $\delta(t)$ is the prediction error. This error is used as an input from the environment as an indication of whether the market is being bullish/bearish.

Furthermore, the updated Q-learning rule which includes the $RW\pm$ is given by:

$$Q_\pi(s_{t+1}, a_{t+1}) = Q_\pi(s_t, a_t) + \begin{cases} \alpha^+\delta(t), & \text{if} \delta(t) > 0. \\ \alpha^-\delta(t), & \text{otherwise}. \end{cases}$$

where $\alpha^\pm$ indicate different learning rates each for the bullish and bearish scenarios respectively.

The network architecture proposed by this model is as shown in Fig. 3.

The adaptive DDPG algorithm includes an actor network and a critic network, similar to the DDPG algorithm. The actor network $\mu(s|\theta^\mu)$ represents the policy learnt by the agent, and after the prediction error $\delta(t)$ is available, the critic network then updates $Q(s, a|\theta^Q)$ according to the prediction error $\delta(t)$ and the learning rate $\alpha^+ = 1$ (or $\alpha^- = 0$), where θ^μ is the set of actor network parameters and θ^Q is the set of critic network parameters. $\mathcal{N}^+$ and $\mathcal{N}^-$ are the random exploration noise added for the positive and negative environment respectively.

Thus, it is said the discussed approach is "adaptive", as it learns to tune the amplitude of it's change based on the type of market (bullish or bearish) observed, and this is done through the learning rate and the exploration noise.

3. Proposed approach

We first provide an overview of our sentiment-aware approach as an extension to the adaptive DDPG algorithm, and discuss the model architecture. We also present our model specification in detail. We then describe the methodology by which we calculate the market sentiment that was used in the model and provide necessary definitions.

3.1. Adaptive sentiment-aware DDPG approach

3.1.1. Model architecture

Our model architecture is as illustrated in Fig. 4.

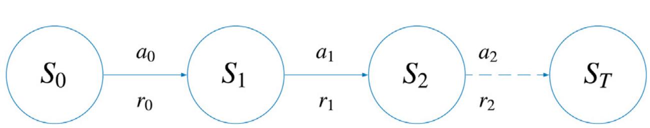

Fig. 2. State transitions in a Markov Decision Process.

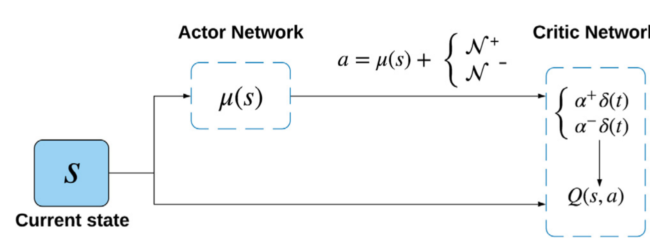

Fig. 3. Adaptive DDPG actor critic network architecture.

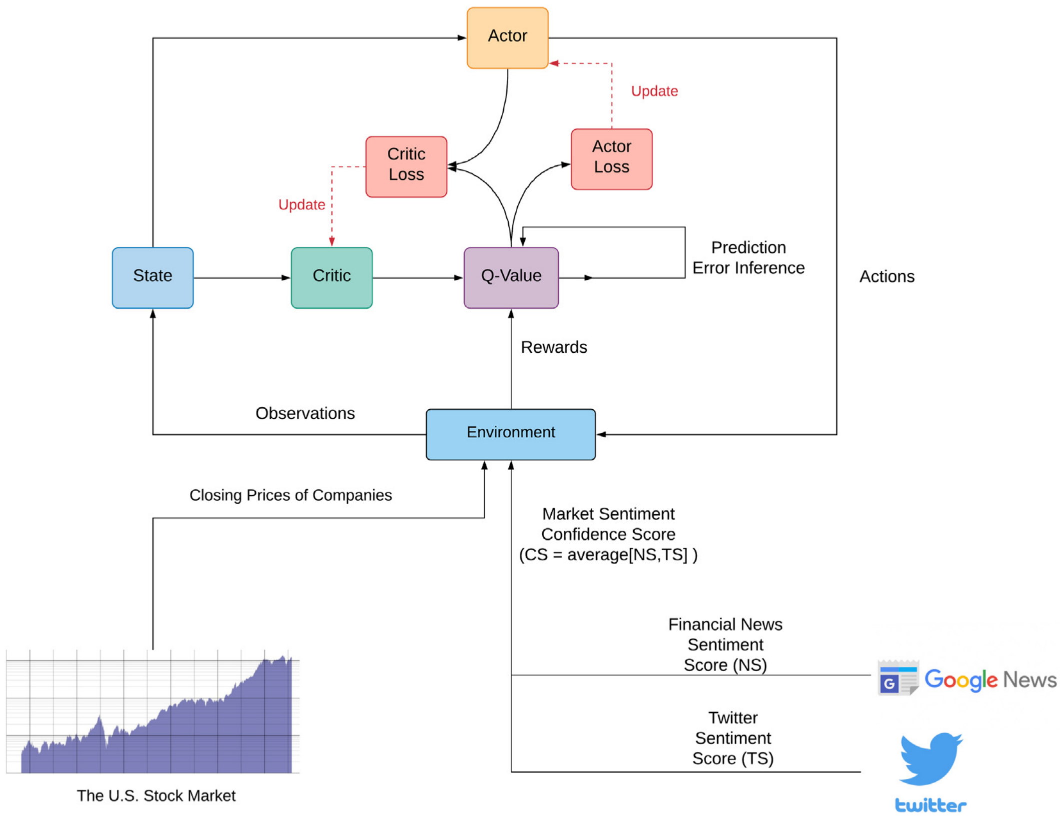

Fig. 4. Adaptive Sentiment-Aware DDPG Actor Critic Framework.

The adaptive DDPG algorithm uses internal prediction error to form indications on the nature of the market. Due to the state-of-the-art results achieved by this algorithm, we build upon it further by tuning in additional external stimulus from the environment that could intuitively be utilized to achieve better trading performance. We extend the state of our trader (agent) by including the confidence score as calculated using the above procedure. We fine-tuned the reward to make our agent learn about the market sentiment effectively.

The architecture is similar to the adaptive DDPG actor-critic framework [48]. As shown in Fig. 4, observations from the stock market environment consist of both the stock price movements and the market sentiments from Google News and Twitter tweets, which are provided to the state of the agent. The actor and critic networks are updated based on the observations and the prediction errors are utilized to control or amplify the updates to the Q-value.

3.1.2. Detailed model specification

We proceed to mathematically detail the formulation of our problem as a finite MDP by specifying the contents and design of the state, action, reward and transition probabilities:

- Each state s in the set of states S is represented as a tuple containing four major components: $\{p_t, h_t, b_t, m_t\}$. At a given day t, p_t stands for the list of closing prices in dollars of the 30 stocks. h_t is the list of the amount(number) of stocks the trader currently possesses as holdings, b_t represents the balance in dollars available with the trader, and m_t represents the lisst of calculated market sentiments of the companies.
- An action a in the set of actions A is a vector containing the quantity of stock that the trader intends to buy, sell or hold. A buy action is indicated by a positive quantity, a sell action by a negative quantity, and a hold action by a zero.

Specifically, for each of the stocks x, the action $a[x]$ would be as follows:

– Buy action: $a[x] \in (0, \beta]$
– Hold action: $a[x] = 0$
– Sell action: $a[x] \in [-\beta, 0)$

where β denotes the board lot – the standardized maximum amount of stock the trader can buy/sell at once.

For example, if the trader wishes to buy 3 Apple stocks and sell 5 Microsoft stocks and hold the remaining stock, then the action would be expressed as $[0 \ldots 0, 3, 0 \ldots .0, -5, 0..0]$ where 3 and -5 occur at indices corresponding to the companies.

- The state transition probability $P_a(s, s')$ is trivially assumed as 1 for all states and actions – i.e., given a state s, if the agent takes an action a, the next state would be s' without any uncertainty involved.
- The immediate reward $R_a(s, s')$ obtained when the trading agent takes an action a at state s when it leads to state s' would be calculated from the change in the value of the asset, along with the scaled market sentiment.

The design of the reward system is crucial for our trading agent to form a robust learning strategy. There are two components in the reward we give to the agent on a day, which are considered together for the overall reward:

– Reward by change in portfolio value from the previous day – if this change is positive, the agent has improved its portfolio value and therefore receives a positive reward as appreciation. Else, it is punished with a negative reward.
– Reward by market sentiment – when the market sentiment is deemed positive, the agent needs to realize that the investor

perspective on the company is good, and thus due to more stock buys the value of the stock i.e. it's price rises. Thus, our trader must buy this company's stock for that day. Similarly, if the market sentiment starts to decline, the agent must then realize that the stock price for the company would start to fall as well, and sell the stock accordingly, at the right time. This strategy is similar to momentum trading, which has been shown to achieve significant positive returns [22].

We now mathematically provide the sequence of steps involved in calculating the immediate reward. Before calculating the reward, we state that the day is t, the agent takes an action a^t that takes it from the current portfolio state s to the next portfolio state s'.

– Step 1: Calculation of $\mathscr{R}_i^t$, which we call the stock-specific sentiment reward of stock i at day t:

$$\mathscr{R}_i^t = \begin{cases} +\eta \times m_i^t, & \text{if } a_i^t >= 0. \\ -\eta \times m_i^t, & \text{if } a_i^t < 0. \end{cases} \tag{4}$$

where $\eta = 1000$ is a scaling factor which we multiply with the market sentiment of the company (a part of the state) m_i^t to ensure a strong enough influence on the reward.
It is noteworthy here to see that $\mathscr{R}_i^t$ would be a positive value if the market sentiment $m_i^t > 0$ and the agent buys or holds the stock ($a_i^t >= 0$), negative otherwise. This is true in other cases vice versa as well, thus reaffirming our intuitive notion on the reward.

– Step 2: Finding the average of all the stock-specific sentiment rewards $\mathscr{X}^t$.

$$\mathscr{X}^t = \frac{\sum_{i=1}^{N} \mathscr{R}_i^t}{N}$$

where $N = 30$ is the number of stocks considered in the portfolio.

– Step 3: Calculating the current state s and next state s' portfolio values ρ_s^t and $\rho_{s'}^t$ as:

$$\rho_s^t = p_s^t \cdot h_s^t + b_s^t$$

$$\rho_{s'}^t = p_{s'}^t \cdot h_{s'}^t + b_{s'}^t$$

where p_s^t is a vector of stock closing prices at state s and day t, similarly h_s^t is the vector of the amount of holdings and b_s^t is the balance amount with the agent.

– Step 4: Computing the actual immediate reward $R_a^t(s, s')$:

$$R_a^t(s, s') = \rho_{s'}^t - \rho_s^t + \mathscr{X}^t$$

3.2. Market Sentiment Confidence Score formulation

For every company on every date, two scores were calculated – google news sentiment score and twitter sentiment score.

3.2.1. Google News Sentiment Score (NS)

Let C be the set of Dow companies in consideration. Let D be the date range in consideration. The google news article headings were obtained $\forall c \in C, \forall d \in D$. Sentiment analysis was performed to obtain a google news sentiment score $NS_{(c,d)}$ $\forall c \in C, \forall d \in D$.

$$NS_{(c,d)} = \frac{\sum_{i=1}^{N} PS(a_{(i,c,d)})}{N} \tag{5}$$

where: $NS_{(c,d)}$ = Google News Sentiment Score for a company c on date d

N = number of articles on company c on date d

$a_{(i,c,d)}$ = i^{th} article published about company c on date d

$PS(a)$ = polarity score of an article a, which ranges from -1 to 1. -1 represents most negative and $+1$ represents most positive.

3.2.2. Twitter Sentiment Score (TS)

Let C be the set of Dow companies in consideration. Let D be the date range in consideration. Tweets were obtained $\forall c \in C, \forall d \in D$. Sentiment analysis was performed to obtain a twitter sentiment score $TS_{(c,d)}$ $\forall c \in C, \forall d \in D$.

$$TS_{(c,d)} = \frac{\sum_{i=1}^{N} w(t_{(i,c,d)}) PS(t_{(i,c,d)})}{N} \tag{6}$$

where: $TS_{(c,d)}$ = Twitter Sentiment Score for a company c on date d.

N = number of tweets on company c on date d.

$t_{(i,c,d)}$ = i^{th} tweet tweeted about company c on date d.

$PS(t)$ = polarity score of a tweet t, which ranges from -1 to 1. -1 represents most negative and $+1$ represents most positive.

$w(t)$ = weight assigned to the tweet t. If the count of likes and retweets for t are 0, the weight is 0. Else, it's equal to 10 times the count of retweets added to the count of likes.

3.2.3. Confidence Score (CS)

Let C be the set of Dow companies in consideration. Let D be the date range in consideration. The confidence score is computed using the financial news sentiment score (NS) and twitter sentiment score (TS) $\forall c \in C, \forall d \in D$.

$$CS_{(c,d)} = \frac{NS_{(c,d)} + TS_{(c,d)}}{2} \tag{7}$$

Apart from the current balance, closing prices of stocks of each company and the amount of holdings our agent has on each of the stocks, the proposed state will also include the confidence score $CS_{(c,d)}$ $\forall c \in C, \forall d \in D$ that the agent trades.

4. Data Acquisition and processing

We now proceed to describe the methods by which we obtain the data required, the different types of data involved, and how this data is processed. We first explain the selection of companies for our portfolio, based on which the data to be acquired is determined. We consider the Dow Jones Industrial Average (DJIA), which is one of the most popular stock market indexes. The DJIA Index is computed on 30 companies listed in the New York Stock Exchange (NYSE), curated based on stock performance, reputation and market value among several other key factors. For the portfolio in consideration by our model, we choose 30 companies that have been or are listed in the DJIA index. In an effort to showcase our model's effectiveness with portfolios comprising varied types of companies, we pick companies from multiple industries including information technology, financial services, retailing and more. The companies, their ticker names as listed in the NYSE, and the industry of their business are as shown in Tab. 1.

Data related to the stock prices of these companies dated from 1st January 2001 to 2nd October 2018 was the dataset required, and it was available in the Compustat database, accessed through Wharton Research Data Services (WRDS). The stocks pricing data was split into training and testing set based on the date range. The stocks pricing data from 1st January 2001 to 30th December 2013 (including 3268 trading days) was used as the training data, and the remaining stocks pricing data from 2nd January 2014 to 2nd October 2018 (including 1190 trading days) was used as the

Table 1
Table listing the 30 Dow Jones Companies in consideration.

Dow Jones Company	NYSE Tickers	Industry
3 M Corporation	MMM	Conglomerate
American Express Company	AXP	Financial Services
Travellers Companies Inc.	TRV	Financial Services
Visa Inc.	V	Financial Services
JP Morgan Chase & Co.	JPM	Financial Services
Goldman Sachs Group Inc.	GS	Financial Services
Apple Inc.	AAPL	Information Technology
Microsoft Corporation	MSFT	Information Technology
Intel Corporation	INTC	Information Technology
IBM Corporation	IBM	Information Technology
Cisco Systems Inc.	CSCO	Information Technology
Boeing Corporation	BA	Aerospace and Defense
Raytheon Technologies Corporation	RTX	Aerospace and Defense
Caterpillar Inc.	CAT	Construction and Mining
Chevron Corporation	CVX	Petroleum Industry
Exxon Mobil Corporation	XOM	Petroleum Industry
McDonalds Corporation	MCD	Food Industry
Coca-Cola Corporation	KO	Food Industry
Johnson & Johnson Corporation	JNJ	Pharmaceutical Industry
Pfizer Inc	PFE	Pharmaceutical Industry
Merck & Co. Inc.	MRK	Pharmaceutical Industry
DuPont de Nemours Inc	DWDP	Pharmaceutical Industry
Walgreens Boots Alliance Inc.	WBA	Pharmaceutical Industry
Walmart Inc.	WMT	Retailing
Home Depot Inc.	HD	Retailing
Nike Inc.	NKE	Apparel
UnitedHealth Group Inc.	UNH	Managed Healthcare
Proctor & Gamble Corporation	PG	Fast-moving Consumer Goods
Verizon Communications Inc.	VZ	Telecommunication
Walt Disney Company	DIS	Broadcasting and Entertainment

testing data. The training and testing split as discussed here is illustrated in Fig. 5.

Along with the stocks pricing data, we had to consolidate text-based data into a dataset that captures the market sentiment of a company on a given date. In order to gather news related market sentiment, we required a corpus of news articles from multiple news sources. Therefore, we used a news aggregator called Google News to obtain this corpus of news articles. It has been shown in numerous studies [53–55] that Twitter can be used as a tool to forecast stock prices and movements. Therefore, we used it as the social media platform for obtaining social media-based market sentiment.

Once the textual data was gathered from Google News and Twitter, it had to be converted into numerical sentiment scores using sentiment analysis. For the purpose of sentiment analysis, we used VADER (Valence Aware Dictionary and sEntiment Reasoner) [56]. VADER is a lexicon and rule-based sentiment analysis tool implemented in Natural Language Toolkit. It is specifically developed for Twitter, with the backdrop of social media in general. VADER has performed well on similar datasets when compared to other sentiment analysis tools and is amongst the best for the purpose of our work [57] since it not only calculates the polarity but also the intensity. VADER calculates a compound score for an input sentence which ranges from -1 (most negative) to $+1$ (most positive). This is referred to as the polarity score for a heading or a tweet in the context of our work.

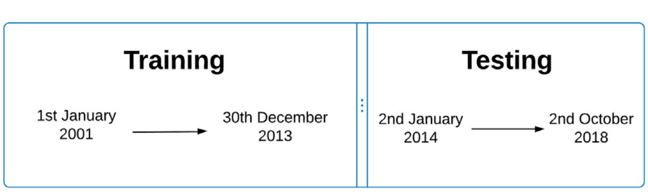

Fig. 5. Division of data into training and testing data based on the timeline.

4.1. Google News

Google News is a powerful and popular news aggregator service developed by Google. It has powerful search functionality to filter news based on keywords, date ranges, relevance etc. We used these parameters to query Google News effectively and retrieve the most relevant articles related to a company. In order to automate the querying of Google News iteratively for all the companies and date range in consideration, we used a browser automation tool called Selenium. Through Selenium, we were able to scrape Google News and gather all the articles for further processing.

We observed empirically that the headings of the articles gave a succinct idea and tone of the article, and were sufficient enough to be used as an indicator of the sentiment of the article towards the company. Therefore, we extracted headings from articles scraped by Selenium. The polarity scores were then obtained by performing sentiment analysis on the extracted headings using VADER. The polarity scores were used to calculate(using Eq. 5) the news sentiment score for the particular company on that date. If there are no available articles for a company on a particular day, the news sentiment score for that day is set to zero, reflecting neutral or unchanged sentiment. The news sentiment score is stored in a file dedicated to storing the news sentiment scores for all the dates for that particular company. The process of Google News scraping using Selenium and sentiment analysis using VADER is illustrated using an example in Fig. 6.

4.2. Twitter

Twitter has the functionality to search tweets based on keywords along with additional search features like date range amongst many others. Obtaining data from Twitter was a 3 step process:

1. Twitter Scraping
2. Data Extraction
3. Tweets Sentiment Analysis

We used keywords based search along with specifying the date to retrieve tweets related to companies on particular dates. We empirically framed keywords in order to get the most relevant tweets. To automate fetching of tweets using the predefined keyword for a company for all the days in consideration, we used Selenium. This was done for all the companies and the scraped tweets were stored in HTML files for each date. This is the Twitter scraping phase.

A tweet consists of multiple data fields such as author, author's handle, the text of the tweet, date and time of tweeting, retweet count, replies count, likes count etc. We refer to retweet count, replies count and likes count as engagement data since these reflect the engagement of the community with the tweet. It's important to observe that when a user retweets another user's tweet, they are echoing the same idea to their followers. Similarly, when a user likes another user's tweet, it usually represents an agreement with the idea or thoughts presented in the tweet. Since the text of the tweet represents the idea or sentiment of the user about the company, it's one of the most important fields. Therefore, we were particularly interested in the date, text and engagement data of the tweets and extracted these fields as part of the tweet data extraction phase.

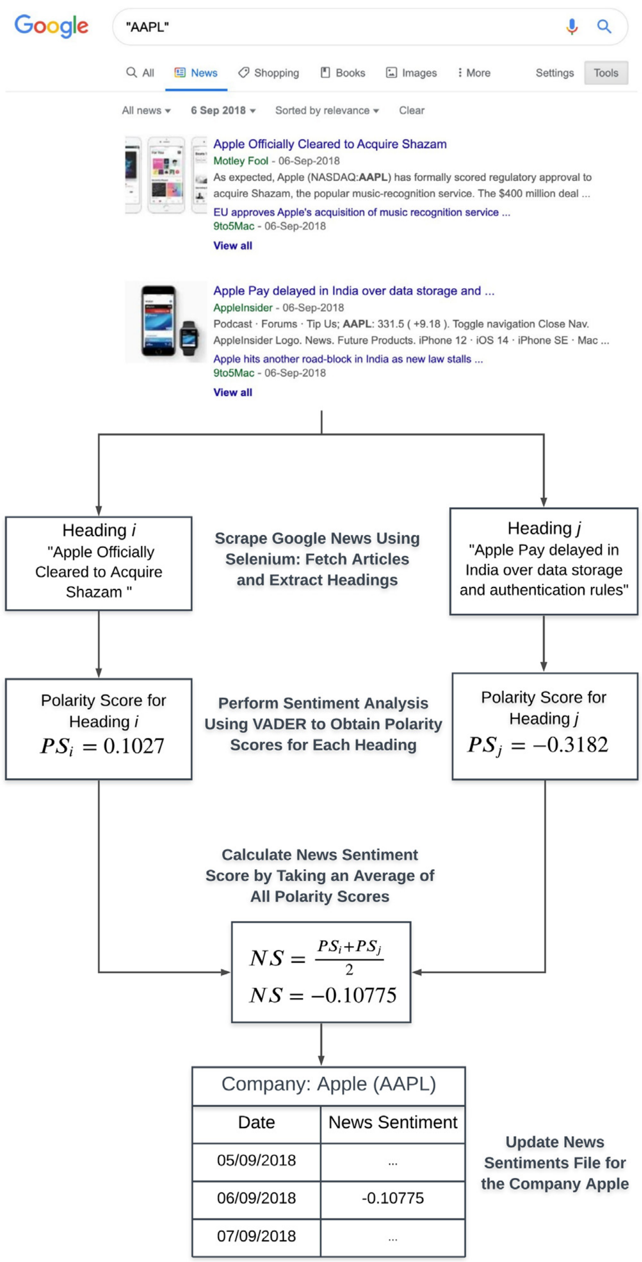

Fig. 6. Working Example With Detailed Steps for Google News.

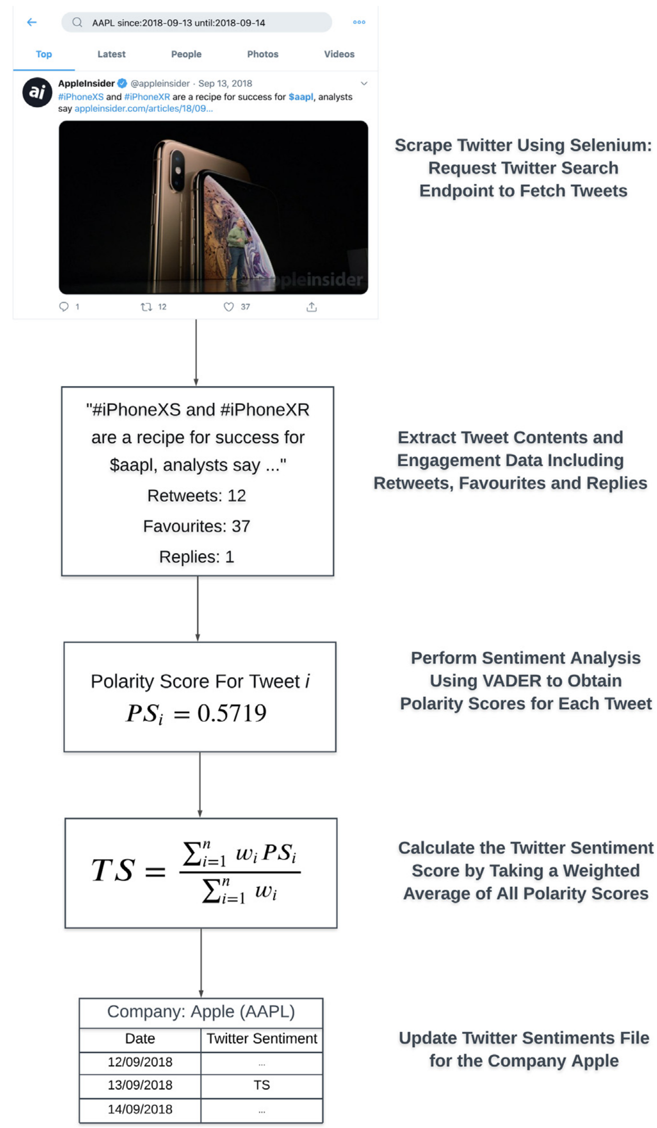

Fig. 7. Working Example With Detailed Steps for Twitter tweets.

Next, we performed sentiment analysis using VADER on texts of tweets and obtained their polarity scores. These polarity scores, along with the engagement data for all the tweets, were used to calculate the Twitter sentiment score (TS) for the company on that particular date using the Eq. 6. If there are no available tweets for a company on a particular day, the tweets sentiment score for that day is set to zero, reflecting neutral or unchanged sentiment. The calculated TS, for the particular date, is stored in a sentiments file dedicated to the company. Such TS is calculated for all dates and all companies. This is the tweets sentiment analysis phase. The process of Twitter scraping using Selenium, tweet extraction using Beautiful Soup and sentiment analysis using VADER is illustrated using an example in Fig. 7.

The sentiment files created from Google News and Twitter were combined to form a final sentiment file which stores the confidence scores, calculated using Eq. 7, for each company. The final sentiment file for each company along with the stock prices data from Compustat database formed the dataset for our work.

5. Results and discussion

Upon testing the learned agent, the series of portfolio values for each day are as plotted in Fig. 8, along with the values obtained by following the baseline approaches of Adaptive DDPG, DDPG and the traditional mean and min variance analyses. The figure is a graph that plots the portfolio value – which measures the worth of the stocks that the trader owns, to the days for which we consider our model's performance against the standard benchmarks. The portfolio value is calculated by adding the value of the stocks owned and the balance amount remaining with the trader agent (if any).

It is observed that the portfolio values achieved by our sentiment-aware approach are consistently higher throughout the considered timeline, compared to the portfolio values obtained when following other baseline approaches. We note that the inclusion of the market sentiment into the adaptive DDPG model has significantly improved the performance of the trader agent.

To assess our model's portfolio allocation, we choose four main metrics – the Sharpe ratio which provides a measure of a unit of return per unit of risk, the annualized return which signifies the quantity of yearly yields from the portfolio, and annualized stan-

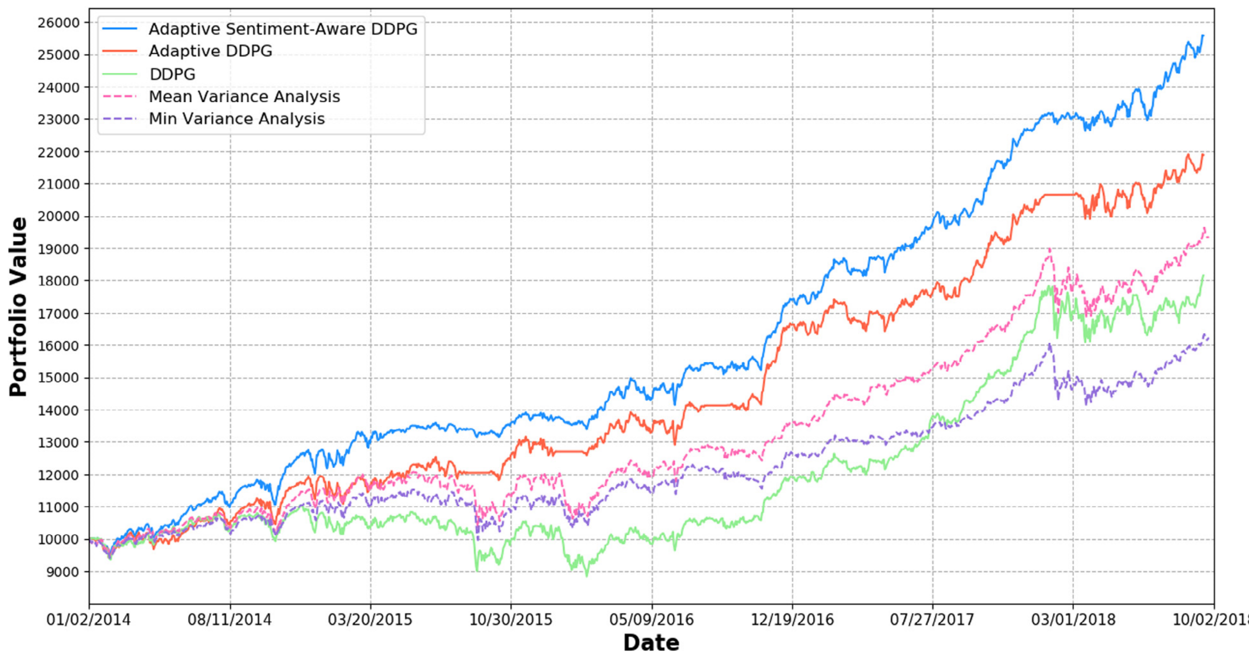

Fig. 8. Plot comparing the portfolio values of our approach and the benchmarks.

dard deviation error which represents the amount of risk taken by the model – which, in investment terms, is a measure of the price volatility, and, the final portfolio value at the last day of trading which measures the eventual return on investment.

We calculate the annualized return by measuring the improvement in the portfolio value of our model over a year. The annualized standard deviation error is the standard deviation of the portfolio values that the agent goes through for all days of the year. This error can be seen as capturing the "steadiness" of the model in terms of delivering returns. The lower this error, the safer our model is for the risk-averse trader. The Sharpe ratio is calculated by dividing the mean of the overall returns with the annualized standard deviation error. (see Tab. 2).

It is noteworthy that our sentiment aware approach has significant improvements across all the considered metrics compared to the baselines. With an initial investment of 10,000 dollars, the final portfolio value of our approach reaches 25,051 dollars which is much higher than 21,881 dollars by Adaptive DDPG and 18,156 dollars by DDPG. Our approach achieves an annualized rate of return of 22.05% as compared to 18.85% and 14.7% returns by Adaptive and DDPG respectively. The risk taken by our model is also lesser compared to the other baselines as shown by obtaining the least annualized standard deviation error of 0.096. With a 2.07 Sharpe ratio value, our sentiment-aware approach is shown to be more robust and effective in balancing return and risk compared to 1.49, 0.93 for the adaptive DDPG and DDPG respectively.

We had observed multiple challenges in effectively evaluating stock portfolio allocation – to begin with, there was an absence of a reliable benchmark dataset [58] upon which results could be sufficiently built upon for comparison with other approaches. And, building a reference dataset for such a purpose from existing datasets was not simple [59] due to multiple formats and lack of adequate information, leading to ambiguity. Added to this, there was no standardization on the evaluation of results – many state-of-the-art papers aimed for different goals for different markets and the methods used to evaluate such goals were also specific to the work done. To address these challenges, our approach has followed a simple dataset format, has choosen a versatile set of companies, and has used standardized metrics for a rigorous evaluation.

6. Conclusion

6.1. Contributions

We highlight our key contributions as follows:

- We've designed a simple and effective methodology for calculating market sentiment given a relevant corpus of unstructured textual data. Instead of restricting the representation of market sentiment into a specific number of categories [37], our approach defines it numerically, i.e., the polarity of the text is expressed as a real number, defined within a range of values. This has a twofold benefit – not only is the sentiment is captured in a more nuanced way, but the utilization of this sentiment in the reinforcement learning process also becomes much more feasible.
- We've specified a learning framework that incorporates the perceived market sentiment into an adaptive deep reinforcement learning algorithm [48]. An agent trained on this framework

Table 2
Tabulating metrics of all approaches.

Approach	Sentiment-Aware ADDPG	Adaptive DDPG	DDPG	Mean Variance	Min Variance
Sharpe Ratio	**2.07**	1.49	0.93	1.25	0.99
Annualized Return (%)	**22.05**	18.84	14.7	15.86	11.48
Annualized Std. Error	**0.096**	0.116	0.147	0.127	0.116
Final Portfolio Value (USD)	**25,051**	21881	18156	19632	16333

can take trading decisions on a daily basis to substantially improve portfolio allocation results. The usefulness of market sentiment in being able to predict stock movements and its ability to influence investor decisions has been the subject of discussion in financial literature [15,19]. In this context, the results we've obtained support the case for market sentiment, indicating its potential for usage in many applications.

- Much of the work on leveraging market sentiment is done on data that is customized for a specific, narrowed-down problem statement. Therefore, existing datasets are either outdated [16], or focus on a small, limited group of companies [17], or fixate over a particular source for obtaining data [18]. As there are various ways of performing sentiment analysis on the obtained textual data, there is a need for standardization of the dataset used [58]. In an effort to overcome these issues, we've built a novel dataset that could serve as a benchmark for further research on market sentiment. Our dataset contains the sentiments of 30 prominent companies listed over the years in the Dow Jones Industrial Average, and is sourced from two of the most prevalent types of platforms for receiving updates on the market – conventional news media and social media, via Google News and Twitter, respectively. Our focus has not been only on the recent years – we've considered a timeline of the last two decades, with every suitable headline and tweet analyzed for each eligible day. We've used VADER [56], an open-sourced lexicon and rule-based sentiment analyzer tool specifically attuned for text on social media, that has performed well compared to other similar tools [57].

6.2. Summary

We've proposed an adaptive, sentiment-aware deep deterministic policy gradients approach to solve portfolio allocation that not only learns from historical stock price trends, but also from market sentiment – which is an influential environment input that captures the overall mood of investors. We've consolidated an extensive dataset of Google News and Twitter tweets that reflect the sentiment of the 30 Dow Jones companies. We've also provided the methodology and mathematical definitions used to calculate market sentiment, and enable the adaptive DDPG algorithm to leverage it sufficiently. Finally, we've shown how our approach works significantly better in comparison with existing baselines in terms of Sharpe ratio and annualized investment return, and discuss some of the issues and challenges that we've come across while establishing a rigid and sound evaluation of the obtained results.

6.3. Future work

Upcoming work could be directed to include more tweets per day, to expand acquisition to gain insights from multiple sources such as stock market-specific news websites (CNBC, BusinessStandard etc.), and to process images as well, as most of the tweets and news online are shared as image snippets in recent times. Multi-agent reinforcement learning approaches can also be explored for this scenario [61]. We also look forward to addressing the presence of multiple exogenous constraints on retail and institutional traders such as transaction costs, trading restrictions, cash holding restrictions and lack of liquidity. Furthermore, it can be stated that natural language processing on financial data is a non-trivial task [60], due to the excessive usage of metaphors, sarcasm, domain-specific terms among other indirect linguistic references in common language, especially in text that expresses an opinion. Being able to comprehend such language would help in estimating market sentiment more effectively.

Funding

The authors received no financial support for the research, authorship, and publication of this article.

Availability of Data and Code

The historical stocks' price data is available via the Compustat Annual Updates database of Wharton Research Data Services (WRDS) here. Additionally, the sentiments data is available in a public repository on GitHub, accessible here.The code used for this study is available in two public GitHub repositories, accessible here and here.

Declaration of Competing Interest

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

References

[1] B.I. Jacobs, K.N. Levy, The complexity of the stock market, J. Portf. Manag. 16 (1989) 19–27, https://doi.org/10.3905/jpm.1989.409244.
[2] H. Markowitz, Portfolio selection, J. Finance 7 (1952) 77–91, https://doi.org/10.2307/2975974.
[3] J. Moody, L. Wu, Y. Liao, M. Saffell, Performance functions and reinforcement learning for trading systems and portfolios, J. Forecast. 17 (1998) 441–470, https://doi.org/10.1002/(SICI)1099-131X(1998090)17:5/6%3C441::AID-FOR707%3E3.0.CO;2-%23.
[4] P.G. Moore, Mathematical models in portfolio selection, J. Inst. Actuar. 98 (1972) 103–148, https://doi.org/10.1017/S0020268100051714.
[5] W.F. Sharpe, A simplified model for portfolio analysis, Manage. Sci. 9 (1963) 277–293, https://doi.org/10.1287/mnsc.9.2.277.
[6] K.J. Cohen, J.A. Pogue, An empirical evaluation of alternative portfolio-selection models, J. Bus. 40 (1967) 166–193, https://doi.org/10.1086/294954.
[7] G.A. Pogue, An extension of the Markowitz portfolio selection model to include variable transactions' costs, short sales, leverage policies and taxes, J. Finance. 25 (1970) 1005–1027, https://doi.org/10.1111/j.1540-6261.1970.tb00865.x.
[8] A.B. Abel, O.J. Blanchard, An intertemporal model of saving and investment, Econometrica 51 (1983) 675–692, https://doi.org/10.2307/1912153.
[9] S. Perrin, T. Roncalli, Machine learning optimization algorithms & portfolio allocation, Mach. Learn. Asset Manag. (2020) 261–328.
[10] C.S. Vui, G.K. Soon, C.K. On, R. Alfred, P. Anthony, A review of stock market prediction with artificial neural network (ANN), in: 2013 IEEE Int. Conf. Control Syst. Comput. Eng., 2013, pp. 477–482.https://doi.org/10.1109/ICCSCE.2013.6720012..
[11] M. Ballings, D. Van den Poel, N. Hespeels, R. Gryp, Evaluating multiple classifiers for stock price direction prediction, Expert Syst. Appl. 42 (2015) 7046–7056, https://doi.org/10.1016/j.eswa.2015.05.013.
[12] M. Abe, H. Nakayama, Deep learning for forecasting stock returns in the cross-section, in: D. Phung, V. Tseng, G. Webb, B. Ho, M. Ganji, L. Rashidi (Eds.), Adv. Knowl. Discov. Data Min., Springer, 2018, pp. 273–284.https://doi.org/10.1007/978-3-319-93034-3_22..
[13] S.M. Bartram, J. Branke, M. Motahari, Artificial Intelligence in Asset Management, CFA Institute Research Foundation (2020), https://doi.org/10.2139/ssrn.3692805.
[14] K. Arulkumaran, M.P. Deisenroth, M. Brundage, A.A. Bharath, Deep reinforcement learning: a brief survey, IEEE Signal Process. Mag. 34 (2017) 26–38, https://doi.org/10.1109/MSP.2017.2743240.
[15] M. Baker, J. Wurgler, Investor sentiment in the stock market, J. Econ. Perspect. 21 (2007) 129–152, https://doi.org/10.1257/jep.21.2.129.
[16] T. Rao, S. Srivastava, Analyzing stock market movements using Twitter sentiment analysis, in: Proc. 2012 Int. Conf. Adv. Soc. Networks Anal. Min., 2012, pp. 119–123.https://doi.org/10.5555/2456719.2456923..
[17] V.S. Pagolu, K.N. Reddy, G. Panda, B. Majhi, Sentiment analysis of Twitter data for predicting stock market movements, in Int. Conf, Signal Process. Commun. Power Embed. Syst. 2016 (2016) 1345–1350, https://doi.org/10.1109/SCOPES.2016.7955659.
[18] M.W. Uhl, Reuters sentiment and stock returns, J. Behav. Financ. 15 (2014) 287–298, https://doi.org/10.1080/15427560.2014.967852.
[19] G.W. Brown, M.T. Cliff, Investor sentiment and the near-term stock market, J. Empir. Financ. 11 (2004) 1–27, https://doi.org/10.1016/j.jempfin.2002.12.001.
[20] J.Y. Campbell, L.M. Viceira, Strategic Asset Allocation: Portfolio Choice for Long-Term Investors, Oxford University Press (2002), https://doi.org/10.1093/0198296940.001.0001.
[21] N. Oliveira, P. Cortez, N. Areal, Stock market sentiment lexicon acquisition using microblogging data and statistical measures, Decis. Support Syst. 85 (2016) 62–73, https://doi.org/10.1016/j.dss.2016.02.013.

[22] N. Jegadeesh, S. Titman, Returns to buying winners and selling losers: implications for stock market efficiency, J. Finance 48 (1993) 65–91, https://doi.org/10.2307/2328882.
[23] W.F. Sharpe, Portfolio theory and capital markets. McGraw Hill Higher Education, New York.https://doi.org/10.2307/2978700..
[24] W.F. Sharpe, The Sharpe ratio, J. Portf. Manag. 21 (1994) 49–58, https://doi.org/10.3905/jpm.1994.409501.
[25] G. Curtis, Modern portfolio theory and behavioral finance, J. Wealth Manag. 7 (2004) 16–22, https://doi.org/10.3905/jwm.2004.434562.
[26] B.G. Malkiel, The efficient market hypothesis and its critics, J. Econ. Perspect. 17 (2003) 59–82, https://doi.org/10.1257/089533003321164958.
[27] Y. Cohen-Charash, C.A. Scherbaum, J.D. Kammeyer-Mueller, B.M. Staw, Mood and the market: can press reports of investors' mood predict stock prices?, PLoS One 8 (2013) 1–15, https://doi.org/10.1371/journal.pone.0072031.
[28] E.F. Fama, The behavior of stock-market prices, J. Bus. 38 (1965) 34–105, https://doi.org/10.1086/294743.
[29] A.W. Lo, A.C. MacKinlay, Stock market prices do not follow random walks: evidence from a simple specification test, Rev. Financ. Stud. 1 (1988) 41–66, https://doi.org/10.1093/rfs/1.1.41.
[30] J.B. De Long, A. Shleifer, L.H. Summers, R.J. Waldmann, Noise trader risk in financial markets, J. Polit. Econ. 98 (1990) 703–738, https://doi.org/10.1086/261703.
[31] J.R. Nofsinger, Social mood and financial economics, J. Behav. Financ. 6 (2005) 144–160.https://doi.org/10.1207/s15427579jpfm0603/4..
[32] A. Shleifer, R.W. Vishny, The limits of arbitrage, J. Finance. 52 (1997) 35–55, https://doi.org/10.1111/j.1540-6261.1997.tb03807.x.
[33] X. Luo, J. Zhang, W. Duan, Social media and firm equity value, Inf. Syst. Res. 24 (2013) 146–163, https://doi.org/10.1287/isre.1120.0462.
[34] E. Cambria, Affective computing and sentiment analysis, IEEE Intell. Syst. 31 (2016) 102–107, https://doi.org/10.1109/MIS.2016.31.
[35] J. Smailović, M. Grčar, N. Lavrač, M. Žnidaršič, Stream-based active learning for sentiment analysis in the financial domain, Inf. Sci. (NY) 285 (2014) 181–203, https://doi.org/10.1016/j.ins.2014.04.034.
[36] W. Antweiler, M.Z. Frank, Is all that talk just noise? The information content of Internet stock message boards, J. Finance 59 (2004) 1259–1294, https://doi.org/10.1111/j.1540-6261.2004.00662.x.
[37] J. Bollen, H. Mao, X. Zeng, Twitter mood predicts the stock market, J. Comput. Sci. 2 (2011) 1–8, https://doi.org/10.1016/j.jocs.2010.12.007.
[38] Q. Li, T. Wang, Q. Gong, Y. Chen, Z. Lin, S.K. Song, Media-aware quantitative trading based on public web information, Decis. Support Syst. 61 (2014) 93–105, https://doi.org/10.1016/j.dss.2014.01.013.
[39] Q. Li, T. Wang, P. Li, L. Liu, Q. Gong, Y. Chen, The effect of news and public mood on stock movements, Inf. Sci. (Ny) 278 (2014) 826–840, https://doi.org/10.1016/j.ins.2014.03.096.
[40] A. Picasso, S. Merello, Y. Ma, L. Oneto, E. Cambria, Technical analysis and sentiment embeddings for market trend prediction, Expert Syst. Appl. 135 (2019) 60–70, https://doi.org/10.1016/j.eswa.2019.06.014.
[41] F.Z. Xing, E. Cambria, R.E. Welsch, Intelligent asset allocation via market sentiment views, IEEE Comput. Intell. Mag. 13 (2018) 25–34, https://doi.org/10.1109/MCI.2018.2866727.
[42] S. Koyano, K. Ikeda, Online portfolio selection based on the posts of winners and losers in stock microblogs, in: P. Bonissone, D. Fogel (Eds.), 2017 IEEE Symp. Ser. Comput. Intell., Honolulu, 2018, pp. 1–4.https://doi.org/10.1109/SSCI.2017.8280902..
[43] L. Malandri, F.Z. Xing, C. Orsenigo, C. Vercellis, E. Cambria, Public mood-driven asset allocation: the importance of financial sentiment in portfolio management, Cognit. Comput. 10 (2018) 1167–1176, https://doi.org/10.1007/s12559-018-9609-2.
[44] E. Cambria, Y. Li, F.Z. Xing, S. Poria, K. Kwok, SenticNet 6: ensemble application of symbolic and subsymbolic AI for sentiment analysis, in: M. d'Aquin, S. Dietze (Eds.), Proc. 29th ACM Int. Conf. Inf. Knowl. Manag. (CIKM '20), New York, 2020, pp. 105–114.https://doi.org/10.1145/3340531.3412003..
[45] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, MIT Press (2018), https://doi.org/10.1109/TNN.1998.712192.
[46] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, M. Riedmiller, Playing atari with deep reinforcement learning, ArXiv (2013).
[47] Z. Xiong, X.Y. Liu, S. Zhong, H. Yang, A. Walid, Practical deep reinforcement learning approach for stock trading, ArXiv (2018).
[48] X. Li, Y. Li, Y. Zhan, X.Y. Liu, Optimistic bull or pessimistic bear: adaptive deep reinforcement learning for stock portfolio allocation, ArXiv (2019).
[49] P. Yu, J.S. Lee, I. Kulyatin, Z. Shi, S. Dasgupta, Model-based deep reinforcement learning for dynamic portfolio optimization, ArXiv (2019).
[50] R. Bellman, A Markovian decision process, Indiana Univ. Math. J. 6 (1957) 679–684, https://doi.org/10.1512/iumj.1957.6.56038.
[51] T.P. Lillicrap, J.J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, D. Wierstra, Continuous control with deep reinforcement learning, in: Y. Bengio, Y. LeCun (Eds.), 4th Int. Conf. Learn. Represent. ICLR 2016 – Conf. Track Proc., San Juan, 2016..
[52] G. Lefebvre, M. Lebreton, F. Meyniel, S. Bourgeois-Gironde, S. Palminteri, Behavioural and neural characterization of optimistic reinforcement learning, Nat. Hum. Behav. 1 (2017), https://doi.org/10.1038/s41562-017-0067.
[53] J. Kordonis, S. Symeonidis, A. Arampatzis, Stock price forecasting via sentiment analysis on Twitter, in: M. Hancock, N. Marquardt (Eds.), ACM Int. Conf. Proceeding Ser., Ontario, 2016.https://doi.org/10.1145/3003733.3003787..
[54] M. Arias, A. Arratia, R. Xuriguera, Forecasting with twitter data, ACM Trans. Intell. Syst. Technol. 5 (2013) 1–24, https://doi.org/10.1145/2542182.2542190.
[55] N. Oliveira, P. Cortez, N. Areal, The impact of microblogging data for stock market prediction: Using Twitter to predict returns, volatility, trading volume and survey sentiment indices, Expert Syst. Appl. 73 (2017) 125–144, https://doi.org/10.1016/j.eswa.2016.12.036.
[56] C.J. Hutto, E. Gilbert, VADER: A parsimonious rule-based model for sentiment analysis of social media text, in: E. Adar, P. Resnick (Eds.), Proc. 8th Int. Conf. Weblogs Soc. Media, ICWSM 2014, Ann Arbor, 2014..
[57] F.N. Ribeiro, M. Araújo, P. Gonçalves, M. André Gonçalves, F. Benevenuto, SentiBench – a benchmark comparison of state-of-the-practice sentiment analysis methods, EPJ Data Sci. 5 (2016) 1–29, https://doi.org/10.1140/epjds/s13688-016-0085-1.
[58] B.S. Kumar, V. Ravi, A survey of the applications of text mining in financial domain, Knowledge-Based Syst. 114 (2016) 128–147, https://doi.org/10.1016/j.knosys.2016.10.003.
[59] A. Khadjeh Nassirtoussi, S. Aghabozorgi, T. Ying Wah, D.C.L. Ngo, Text mining for market prediction: a systematic review, Expert Syst. Appl. 41 (2014) 7653–7670, https://doi.org/10.1016/j.eswa.2014.06.009.
[60] E. Cambria, S. Poria, A. Gelbukh, M. Thelwall, Sentiment analysis is a big suitcase, IEEE Intell. Syst. 32 (2017) 74–80, https://doi.org/10.1109/MIS.2017.4531228.
[61] W. Bao, X.Y. Liu, Multi-agent deep reinforcement learning for liquidation strategy analysis, ArXiv (2019).