International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 /doi:10.14445/22315381/IJETT-V70I1P205

Volume 70 Issue 1, 43-53, January, 2022

Deployment of Deep Reinforcement Learning and Market Sentiment Aware Strategies in Automated Stock Market Prediction

Keerthi Sagiraju[1], Prof. Shashi Mogalla[2]

[1,2]*Department of CS&SE, Andhra University College of Engineering, Vishakapatnam, India.*

[1]keerthisagiraju86@gmail.com,[2]smogalla2000@yahoo.com

Abstract - *Tracking and responding to the dynamic stock market to maximize profit within a risk-controlled framework to meet the investment objective continues to be challenging. In addition to this, it is understood that market sentiment also plays a pivotal role in investment decisions. The deep reinforcement learning concept shows promising outcomes on stock market prices by training an intelligent agent. The stock market investments and returns can be predicted using Reinforcement learning with historical data and agent-based training in the given environment. In the proposed approach, a deep reinforcement learning agent is trained that uses historical stock and market sentiment consisting of Dow Jones and S&P 500 and get the resultant strategy in trading with Auto Encoder and LSTM along with four algorithms, namely Advantage Actor-Critic (A2C), Proximal Policy Optimization (PPO), Deep Deterministic Policy Gradient (DDPG) and Deep Q-Learning (DQN). These are compared using the financial metrics, Sharpe ratio, Sortino ratio, max drawdown, cumulative return, annual volatility, and annualized investment return.*

Keywords - *Deep reinforcement learning (DRL), Stock Market Trading, Sentiment Analysis, Auto Encoder(AE), Long Short-Term Memory(LSTM).*

I. INTRODUCTION

The outbreak of COVID – 19 triggered a big fall in share prices of the stock market. Since the globe has changed economies, people's lifestyle and the future of business is reflected in the ups and downs of the share market [1]. In Data Mining, one of the interesting fields is stock price prediction. The stock market prices are affected by many variables in particular factors like COVID-19 that are very unpredictable. The stock market is unstable [2]and intricate due to a sequence of local and universal “black swans” in 2020 like COVID-19. The studies reflect the deficiency of acceptable models which forecast the variations of the stock prices in such unpredictable conditions.

Stock trade can be automated using deep reinforcement learning [3]. Reinforcement Learning [4] uses an agent that gets trained by performing actions based on the conditions it is subjected to in the given environment to get the maximum reward, thus needing no labeling like supervised learning [5,6]. It performs very well with less historical data. The volatility, nonlinear, random, no stationary, and noisy features of the stock market have made it more demanding and challenging to predict the stock price.

The sections of this paper are organized as follows: Section 2 deals with the Literature Survey. Section 3 provides the proposed approach. Section 4 talks about the details of the experimental setup. Section 5 discusses the experimental results. Lastly, Section 6 is the summary of the work.

II. LITERATURE REVIEW

Eduardo Jabburet. Al in [5] used the original and authentic dataset of the Brazilian Stock Exchange to emphasize the trading process in volatile and High-Frequency market scenarios. In their work, the device and assess some models of agents for the stock market. Their system uses GA for the operation of rough set analysis while at the same time examining the max and the near max cut-points of the discretized data to find the suboptimal or optimal trading rules. Jae Won Lee, in [7], proposed a method of applying RL to the problem of stock price prediction by using the Markov process and is optimized by RL based algorithm. Here, an RL algorithm is adapted, and the function estimation by ANN is used to evaluate the values of each state, which reflects the trend of the stock price at a given time.

Christopher Chou et al. in [8] proposed an approach that integrates the investors' sentiments, the historical data of stock prices, and the observed mechanism to forecast stock prices. To combine sentiments of the investors' tweets and categorize them using the sentiment indexes, which helps in buying or selling. An LSTM is created to predict stock values. Christopher Chouet. al [9] distinguished the performance of the forecasting models by taking into count the COVID-19 affecting factors on the stock market prediction.

In the recent trend, RL-based algorithms have been explored for stock market prediction. Li, Y. et al. [10] proposed the DRL technique to prove the reliability and availability in stock prediction. This approach proved the feasibility of DRL for credibility and financial markets. Carta, S. et al. [11] proposed a multi-ensemble multi-layer stock trader application that consists of a stacking preprocessing layer, RL-based meta learner layer, and the final ensemble layer. Rundo, F. [12]proposed an algorithm comprising both RL and supervised Deep Learning (DL) for predicting the short-term trend in FOReign Exchange (FOREX) stock market to increase the return on investment (ROI) in a high-frequency trading (HFT) algorithm.

Long, W. et al. [13] developed a multi-filters neural network for automatically extracting features from stock time-series data. Furthermore, convolutional and recurrent filters are combined for the feature extraction mechanism. Dang, Q. V.et al. [14] employed a simple strategy of greedy baseline algorithm for the prediction stock price using a supervised DL algorithm called Recurrent Neural Network-Long Short Term Memory (RNN-LSTM). Jia, W. U., et al. [15] developed an agent-based on LSTM network for learning the temporal pattern in the dataset and for automatically trading according to the historical data and the current condition of the market. Huotari, T. et al. [16]proposed a Convolutional Neural Network (CNN) based DRL agent for portfolio stock selection management for S&P 500 stock data of 21 years.

Thus the RL has shown state-of-the-art performance in stock market prediction. Hence, to improve the performance of DRL in stock prediction, this research proposes a DRL scheme with a hybrid Autoencoder LSTM network as the agent in state observations.

III. PROPOSED MODEL

To create profound and consistent features ahead of stock price to predict, the research gives a DLframework to financial time series utilizing a DL-based predicting method that incorporates both autoencoders and LSTM. Figure 1 demonstrates the workflow of this framework. The complete approach of each module is presented as follows.

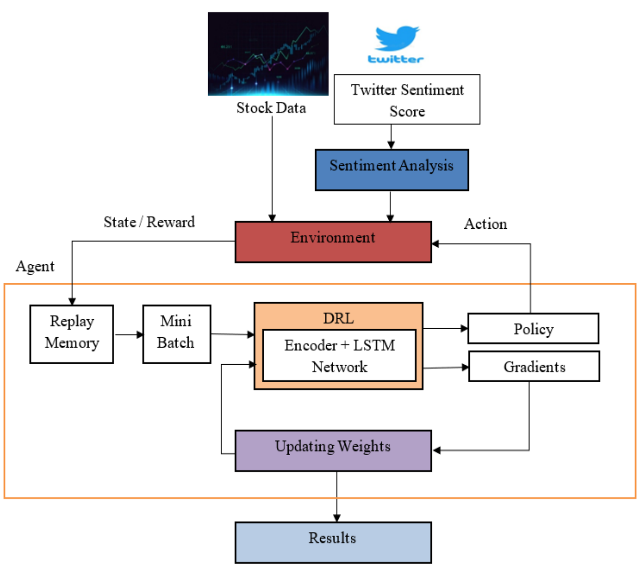


Figure 1: Proposed Framework

A. Dataset Used

Daily historical closing prices (2006-2021) for Dow Jones and S&P 500 were collected from Yahoo Finance to form the dataset. In this project, for training 2006-2016, data are selected, and for testing 2016-2021 dataset is used to perform Sentiment Analysis.

B. Twitter Sentiment

Analysis in sentiment analysis[17], the sentiment score of any statement obtains a meaningful opinion from the text, based on which the nature of the judgment is classified as positive, negative, and neutral. In this paper, Valence Aware Dictionary and Sentiment Reasoner (VADER), shown in Figure 2, is a lexicon and rule-based sentiment analysis tool is used to sense the sentiments shown in social media [18].

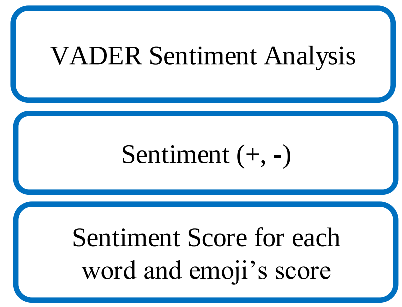


Figure 2: VADER Sentiment Analysis

The count of retweets, replies and likes constitutes the engagement data; the date and text are the predominant Tweet's data and are extracted.[19] Gives the relevant theory to emphasize the effect of sentiment [20] on traders while investing in stock market-related decisions. The main point of evaluating sentimental analysis is to identify whether the sentence is subjective or objective. If objective, nothing is needed; else, if it is subjective, then its polarity (nonpartisan, negative, positive) and their quantities of emojis are verified in the tweet.

C. Stock Market Environment

To train the agent in a simulated real trading environment so that it interacts and learns is the first step. Secondly, a deep reinforcement trading agent is trained. In trading, a variety of information requires to be considered, for instance, the stock prices data, present holding shares, technical indicators, etc.

D. Markov Decision Process (MDP)

The Markowitz model becomes infeasible as the number of speculations raises because it is written as a quadratic program. MDP assigns weight to each asset's Markov state. The present weight invested and the economic state of all assets are stored in each state of the MDP. MDP[21] modeling application for financial markets has been widely studied. MDP can be described as a sequential decision issue and can be utilized to decide the probability distribution on states. The outcome obtained in MDP entirely depends on the action and current state of the agent. The main components of MDP are as follows:

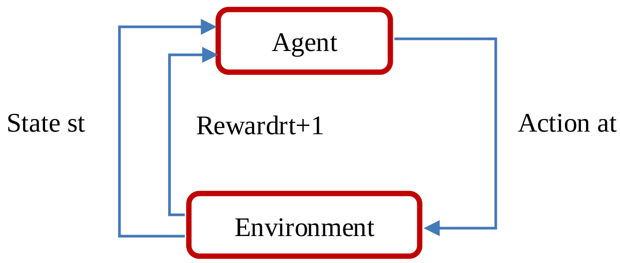


Figure 3: Components of MDP

In figure 3, at each time step, the state is given by the environment, and the agent takes action according to the state and receives the reward and next state. The (s, a, r, s') are saved in the replay memory. For Optimization at each iteration, a batch of (s, a, r, s') is selected for training. The replay memory has a specific capacity filled after which a random (s, a, r, s') is substituted with new (s, a, r, s').

E. Long Short-Term Memory and Auto Encoder

Long-term reliance data are remembered efficiently in LSTM networks which is the altered variety of recurrent neural networks (RNNs). RNNs[22] come with a diminishing gradient problem, and this problem is solved in LSTM networks. The key of an LSTM network is the memory unit. The input and output gates control the memory units' input and output data, respectively. LSTM is a similar technique used for forecasting stock prices and with NNs predicts continual values. LSTM is known for its strength to retain long-term memory [23].

The inputs and the outputs are the same for autoencoders that belong to NNs. The output is redesigned after compressing the input into latent space format. The idea of constructing autoencoders is done by duplicating the input to get the output and to learn the data's properties. There is only approximate duplication to match the training data.

AutoEncoder + LSTM Process:

1. Training the autoencoder.

2. To generate attributes using encoders.

3. Forecast-based LSTM is trained in the last step to furnish the prediction of the modified closing price.

[24] The study gives a novel DL framework autoencoders (AEs), and LSTM is combined for stock price prediction. The DL framework consists of two steps. First, SAEs are applied to generate deep high-range attributes to detect stock prices. Second, high-range denoising attributes are given into LSTM to estimate the next day's closing price. The six market indices and their respective index features are selected to record the efficiency of the proposed method. Results proved that the proposed method outperforms the remaining methods both in profitability performance and predictive accuracy.

F. Gradient descent

The model finds a predicted value throughout the training iterations. The error amongst the predicted and actual values is defined as the error value related to the loss function or cost function of the gradient descent algorithm. The main objective is to increase investment performance and decrease the risk.

G. Performance Metrics

The metrics used to compare the performances of stock market [22] analysis and twitter stock market for DJIA and S&P_500 are the following. The Sharpe ratio is a risk-adjusted return statistic for a financial portfolio. By the accurate Sharpe ratio value, the top-performing specialist of PPO, A2C, DQN, and DDPG can be chosen.

$$SharpeRatio = \frac{(R_p - R_f)}{\sigma_p} \qquad (1)$$

Where, R_p = return of portfolio, R_f=risk-free rate, σ_p = portfolio's excess return standard deviation.

The Sortino ratio is similar to the Sharpe ratio that penalizes only those returns that fall under the threshold of a specific user-defined target and unlike the Sharpe ratio that penalizes both that fall above and under the threshold. The Max drawdown gives the downside risk over a specific time period. The annualized return is the amount of money earned on an average with a yearly investment over a certain period of time. The cumulative return is nothing but the aggregate or total return that is not a yearly investment. The next measure is the annual volatility which is the product of the standard deviation with a square root of 252 considered annually.

H. Deep Reinforcement Learning (DRL) Algorithms

In our proposed system, the Deep RL algorithms (DRL), i.e., DQN, DDPG, PPO, and A2C, are applied along with autoencoder and LSTM. The results after applying the algorithms are compared and observed.

a) Deep Q-Networks (DQN)

To deal with the restrictions of Q-learning[23], DQN forms two-dimensional clusters with the help of a Neural Network. Q-learning is a model-free RL algorithm. DQN[24] uses a Neural Network to assess the Q-learning work. Q-learning agents can't estimate value for states that aren't visible. By introducing Neural Network, DQN resolves this issue which restores the two-dimensional array. The expert replay function can resolve the problem of dynamic distribution. The enhanced formula of Q-Learning is given below:

$$Q'(s,a) = Q(s,a) + [r + \gamma max_a . Q(s',a') - Q(s,a)] \qquad (2)$$

The DQN loss function is as follows:

$$L(\theta) = E\,[(TargetQ - Q(s,a;\theta))^2] \qquad (3)$$

$$TargetQ = r_{t+1} + [\gamma max_a . Q(s', a'; \theta)] \quad (4)$$

Where, ‘a’ is the agents’ action, ‘s’ the state, ‘r’ the reward, and a′, Q′, s′is the updated value of a′, Q′, s′.

b) Deep Deterministic Policy Gradient

Deep Deterministic Policy Gradient (DDPG) [23] uses slow-learning and Experience Replay target networks from DQN, and it relies on DPG, which can operate over consistent activity spaces and hence model-free. It merges the features of DQN and Deterministic Policy Gradient and DQN (Deep Q-Network). It uses slow-learning and Experience Replay target networks from DQN, and DPG, which can operate over constant activity spaces. It utilizes off-policy information and the Bellman condition to learn the Q function. The critic network is renovated by lowering loss function L(θ^Q). It is nothing but the difference expected of both critic network Q and target critic network Q′i.e.,

$$L(\theta^Q) = E_{s_t a_t r_{t+1}} \sim buffer[y_t - Q(s_t, a_t|\theta^Q))^2] \quad (5)$$

c) Proximal policy optimization

Proximal Policy Optimization (PPO)[26] is an effectual reinforcement learning which resolves consistent control behavior. It is a strategy-based technique. First request enhancers like Gradient Descent technique can be utilized to upgrade the outcome. This is as follows:

$$L^{PG}(\theta) = E_t\left[log\pi\theta\left({}_{a_t}|{}_{s_t}\right) * A_t\right] \quad (6)$$

$\boldsymbol{L}^{PG}$ (θ)= policy loss, Et is expected, log ∏θ (a_t| s_t) is log probability of taking that action at that state, A_t is the advantage if A>0, this action is better than the other action possible at that state.

d) Advantage Actor-Critic(A2C)

The Q value can be learned by parameterizing the Q function with a deep neural network. The *Actor-Critic* (A2C) [27] the algorithm encompass "critic," which assesses the value capacity, which could be state or activity value respectively (Q, V values) and "actor" that renew the policy scattering toward the path recommended by the critic. The abilities of the actor and the critic are defined using deep NNs construct using different layer perceptron. The critic neural organization defines the Q value; thus, it is referred to as *Q Actor-Critic*. This can be formulated as below:

$$\nabla_\theta J(\theta) = E_t[\Sigma_{t=0}^{T-1} \nabla_\theta \log \pi_\theta \left({}_{a_t}|{}_{s_t}\right) G_t] \quad (7)$$

Where J is the gradient, E_t, the expected reward at time t, ‘a’ is the actions by an agent at time instance ‘t’ and ‘s’ is the state where the agent is present at time instance ‘t’.Better returns are expected with the quantity of financing risk[28] as the Sharpe ratio of the agent is higher.

IV. EXPERIMENTAL SETUP

In this section, the elements used to explore our proposed model for algorithmic stock trading.

A. Data Acquisition and processing

The Dow Jones Industrial Average (DJIA), S&P_500 are familiar stock market indexes taken in this work. The data is collected from yahoo finance only from 2006 to 2021; the below figure illustrates the break-up of data into train and test with respect to time.

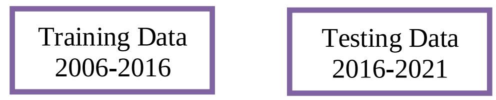


Figure 4: Division of data into train and test date depending on the timeline

The text-based data and the stock pricing data, which includes the training and the testing data, are merged into the dataset so that the companies' market sentiment score is consolidated. In the proposed system, twitters sentiments are used to estimate the volatility of the stock market. VADER computes an aggregated value for an input tweet that scales from −1 to + 1. The counts of retweets, replies, and likes are referred to as engagement data about the company, and these represent the sentiment of the user.

Daily historical closing prices (2006-2021) for Dow Jones and S&P 500 were collected from Yahoo Finance to generate the dataset for training 2006-2016 data are selected and for testing 2016-2021 data that is also considered as test data to conduct Sentiment Analysis. The model consists of LSTM and GRU. Figure 5,6. LSTM with AUTO-ENCODER illustrates the hybrid method with the integration of sentiment analysis and DRL for the forecasting of S&P 500, Dow Jones, and Twitter stock data is to understand the long and short-term effects of news events on the index. 15 years of data can be considered (Till August 2021). The training-testing split should be 70% and 30%. Figure 6: represents the stock market as a graph that indicates close value and dates for the whole stock market data of Twitter. Figure 7 illustrates S&P_500 stock market - Graph indicates close value and dates for whole stock market data of Dow Jones and S&P 500 is the index trade market stalking the efficiency of five hundred big organizations recorded trade market. These drivers play an important role in the overall market sentiment, which is our specific interest.

Created DRL environment

1. Imported necessary libraries

2.Installing tensor flow -gpu==1.15.0

3. Installing stable -baselines

4. Installing gym-any trading environment

5. Reading CSV files through pandas

6. The date is converted to a date-time model.

7. Through gym. Make command the stocks in the dataset are loaded.

B. Experimental Methodology

The preprocessed data is considered as the input. In the construction step, the remaining data is used, i.e., the testing data, to explore the output. This includes the past recent pandemic time data that ran out of profits.

1. Created Action, Reward, and profit
2. Based on the action_space.sample in env, the action is calculated
3. From env. Step (action) rewards and info are calculated.

The states and actions are given as input to the LSTM autoencoder. From this, the training and validation losses are calculated. The DQN agent is used for creating an agent and calculating the total losses and total profits. Figure 8 shows the total losses and total rewards calculated based on the DQN agent.

V. EXPERIMENTAL RESULTS

Comparison of performance and evaluation metrics of DJIA+Twitter Stock Market represented in the table illustrates a comparison of Algorithms for DJIA with Tweets. The DQN and DDPG algorithms provided relatively good outcomes, though the DDPG policy gradient scheme produced enhanced results as it merges the features of Deterministic Policy Gradient (DPG) and Deep Q-Network (DQN). The separate score data is obtained for both DJ and S&P500, and comparison charts are drawn separately for -with sentiment and without sentiment analysis for the period of 2016 to 2021. The major challenging thing in this current approach is Large State Space. There is tremendous growth in the action, state pair space since the number of assets in this portfolios' benefits get increased. There is an exponential growth with the action space as the assets' number increments, thus decreasing the system's performance.

DJIA Company stocks are higher as compared to S&P 500. It was seen through the comparison charts figures 5, 6 with and without sentiment [29] score analysis. Autoencoder, which is an unsupervised ANN [30], can prepare a compressed encoded form of data and can learn to reconstruct it. Initially, after training the autoencoder, the encoder part acts as the attribute generator. The last step is to train the LSTM based predictor, such that the modified closing price forecasting for the next day is furnished.

A. DJIA stock market

Based on the agent's performance mentioned in Table 1 and Figure 5, it was observed that the DQN agent is extra vulnerable to risk.

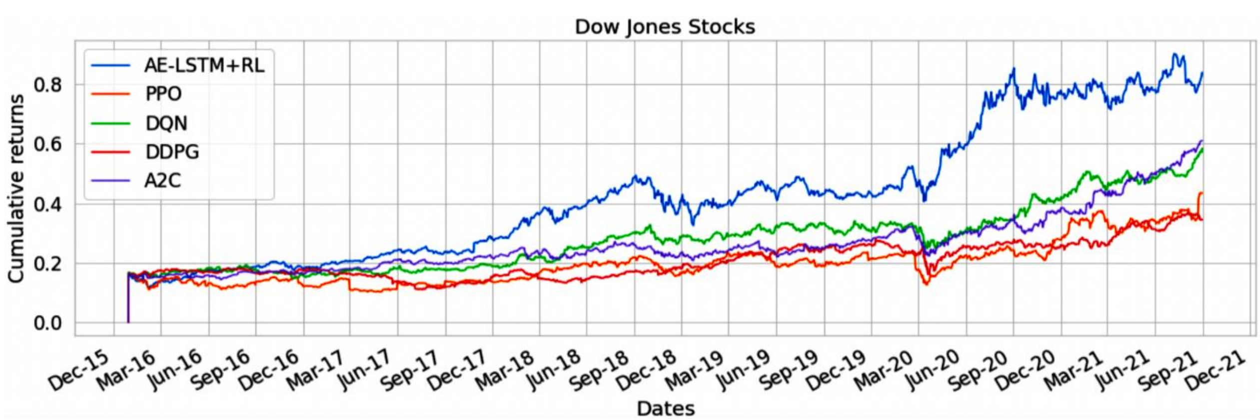


Figure 5: Comparison of Algorithms for DJIA Stock Details

It has the lowest yearly irregularity of 11% and max drawdown −0.30% out of five agents. Therefore, DQN is best at handling a DJIA stock market. The proposed AE-LSTM+RL agent has the highest aggregative return, 72.0% out of the five agents. Both A2C and DDPG accomplish alike but not as best as AE-LSTM+RL, and it has less annual return 13% and cumulative returns of 64% and 68%. Since PPO has a higher annual Return among the other five agents and it is treated as a reciprocal strategy at a DJIA stock market.

B. DJIA stock market-based sentiment

Analysis of Agent Performance based on [24]sentiment analysis: based on the data incorporated in Table 1 and Figure 5,6, it is identified that DJIA stock market-based sentiment has higher Annual return ranges, and the DDPG agent is extra vulnerable to risk. It has the lowest yearly irregularity, 13%, and max drawdown −0.36% out of five agents. Therefore, the DDPG agent is best at dealing with a DJIA stock market based on sentiment analysis.

TheSENTIMENT_AE-LSTM+RL has the greatest cumulative return of 62%, an annual return of 17% among the five agents. So, the SENTIMENT_AE-LSTM+RL agent is exhibited when dealing with a DJIA stock market based on sentiment analysis. Since both the A2C agent and DQN agent have higher annual returns but with fewer cumulative returns of 25% and 22% so they can't be used as a complementary strategy at the stock market.PPO agent has a very less cumulative return of 11% as compared to all other five agents.

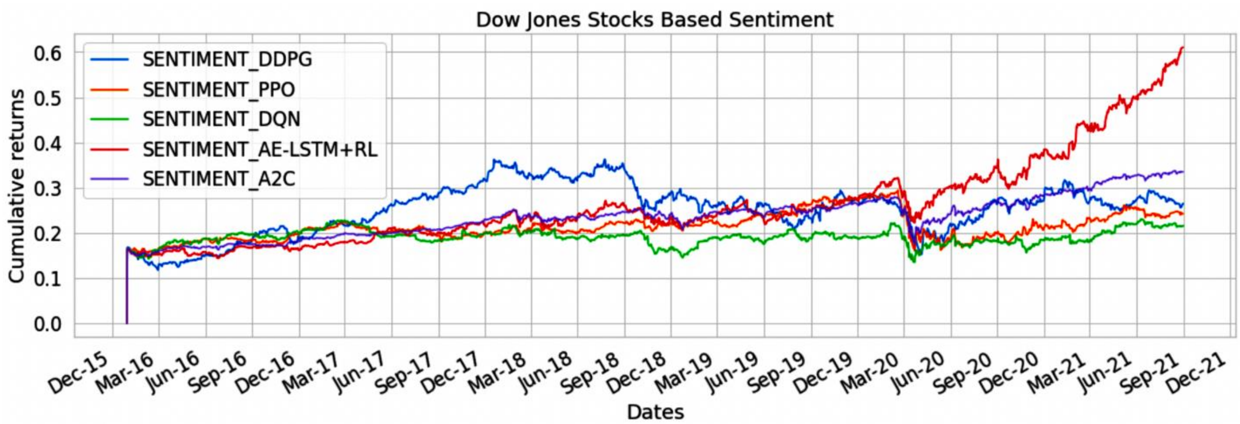


Figure 6: Comparison of Algorithms for DJIA With Tweets

C. S&P_500 stock market

Based on the agent performance mentioned inTable 1 and Figure 7, it was observed that the PPO agent is extra vulnerable to risk. It has the lowest yearly irregularity of 12% and max drawdown −0.19% out of five agents. Therefore, PPO is best at dealing with the S&P_500stock market. A2C agent is best at the following trend and performs well at producing the best returns; it got the greatest cumulative return, 81.0%, and an annual return of 17% out of five agents. Therefore, an A2C agent is preferred when facing an S&P_500 stock market. AE-LSTM+RL has slightly less value of cumulative return as 78% compared to A2C agent. Both DQN and DDPG performs similarly and has fewer annual return with ranges of 15% and 12%, and these two agents have cumulative returns of 42% and 35%. Since A2C has a higher annual Return among the other five agents, it can be used as a complementary strategy at an S&P_500 stock market.

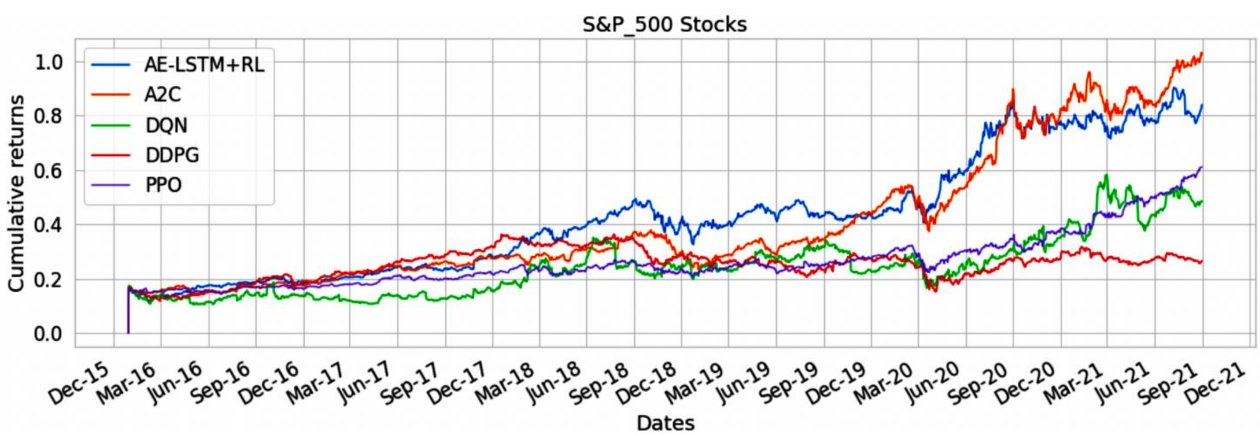


Figure 7: Comparison of algorithms for S&P_500stock data

D. S&P_500 stock market-based sentiment

Analysis of Agent Performance based on sentiment analysis: From Table 1 and Figure 8, it is observed that S&P_500 stock market-based sentiment has higher Annual volatility ranges, and the DQN agent is extra vulnerable to risk. It has the lowest yearly irregularity, 11%, with a max drawdown of

−0.01% out of the five agents. Therefore, the DQN agent is best in handling the S&P_500 stock market based on sentiment analysis. SENTIMENT_AE-LSTM+RL agent is best in the following trend and performs well at producing the best returns; it got the lowest cumulative return, 70.0%, and an annual return of 17% among the five agents. So, the SENTIMENT_AE-LSTM+RL agent is preferred when facing an S&P_500 stock market based on sentiment analysis. Since both PPO agent and DDPG agent have higher cumulative returns of 52%, 46% as compared to A2C and DQN agent, so, these agents can be used as a complementary strategy in the stock market.

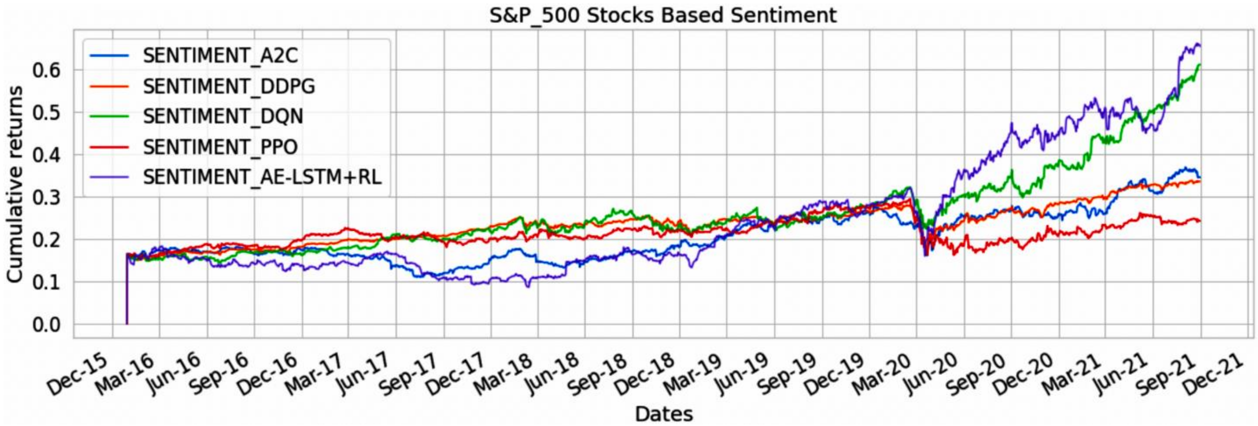


Figure 8: Comparison of Algorithms for S&P_500 with Tweets

E. Score Comparison

Comparison of SRs for both DJIA and S&P_500:

Standard Comparison: Fig.5reveals that the AE-LSTM+RL model strategy significantly outperforms DJIA and S&P_500 stock data. From Table 1, it is evident that the scheme gets aSR 1.75 for DJIA and 1.54 at S&P_500 and is greater than the SR of other agents such as A2C, PPO, DQN, and DDPG. The annualized return of the AE-LSTM+RL model is also greater; the annual irregularity is at the bottom-most, recommending that autoencoder with LSTM strategy beats the other type of agents in balancing risk and return. The AE-LSTM+RL model strategy outperforms A2C with an SR of 1.02, PPO with an SR of 0.64, DQN with an SR of -0.94, and DDPG with an SR of 1.08, separately at the DJIA stock market. AE_LSTM+RL model strategy dominates A2C with an SR of 1.28, PPO with an SR of 1.04, DQN with an SR of 1.54, and DDPG with an SR of 1.28, separately at the S&P_500 stock market.

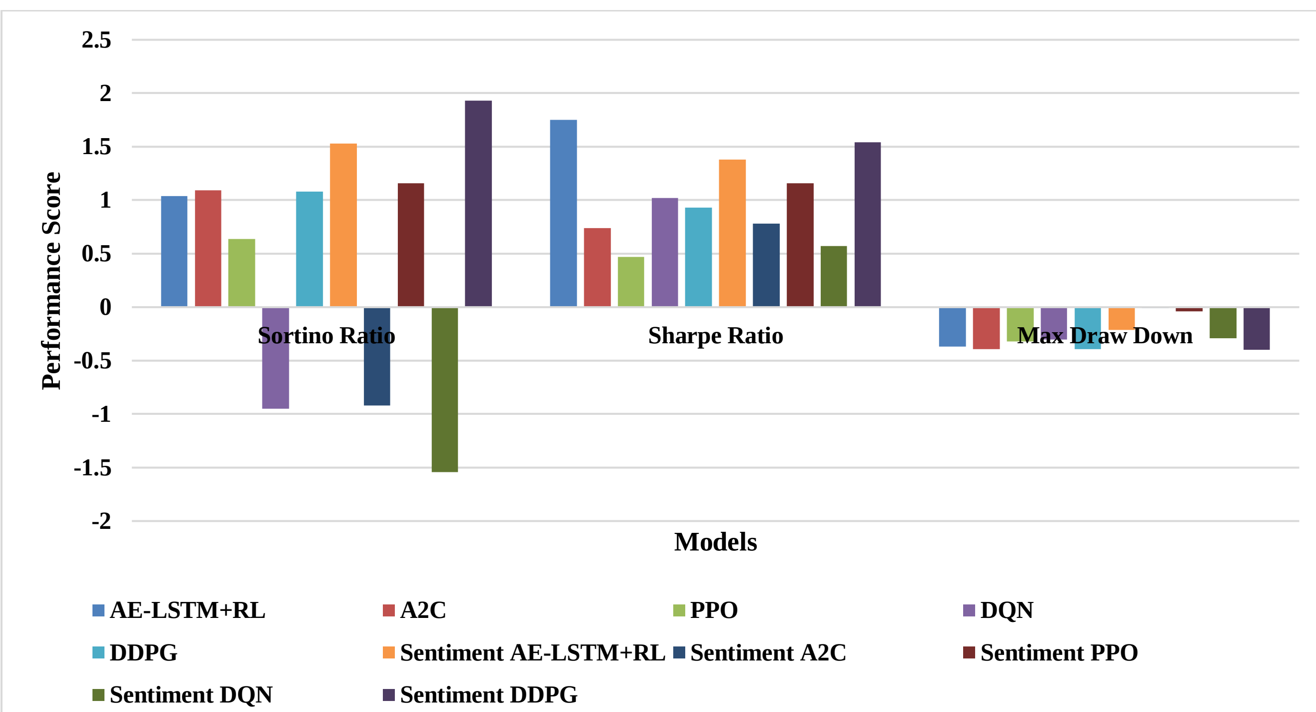
F. Comparison of Sharpe ratios for both DJIA and S&P_500 on sentiment

Standard Comparison: Figure 6reveals that our SENTIMENT_AE-LSTM+RL model strategy significantly outperforms DJIA and S&P_500 stock data. From Table 1, the schemegetsaSRof 1.38 for DJIA and 1.85 at S&P_500 and greater to the SR of other agents such as A2C, PPO, DQN, and DDPG.

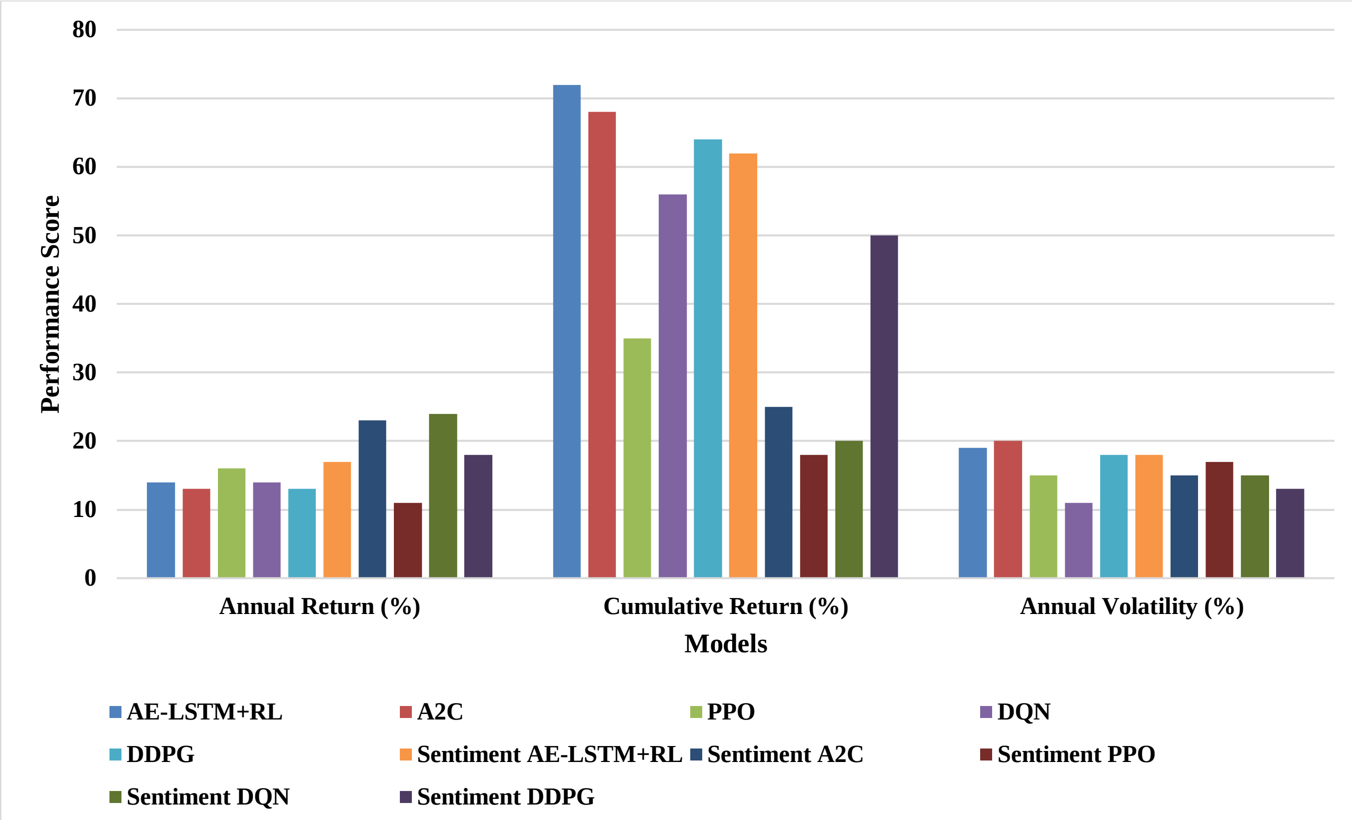
The annualized return of the SENTIMENT_AE-LSTM+RL model is also greater; the annual irregularity is at the bottom-most, recommending that autoencoder with LSTM strategy beats the other type of agents in balancing risk and return. SENTIMENT_AE-LSTM+RL model strategy exceeds A2C with an SR of 0.78, PPO with an SR of 1.16, DQN with an SR of 0.57, and DDPG with an SR of 1.54, separately at the DJIA stock market. SENTIMENT_AE-LSTM+RL model strategy dominates A2C with an SR of 1.41, PPO with an SR of -0.92, DQN with an SR of 0.98, and DDPG with an SR of 1.86, separately at the S&P_500 stock market. Therefore, the proposed SENTIMENT_AE-LSTM+RL strategy outperforms the four individual algorithms.

TABLE.1:Comparison of performance and evaluation metrics for DJIA-STOCK MARKET, DJIA+TWITTER STOCK MARKET, S&P_500 STOCK MARKET, S&P_500 TWITTER STOCK-MARKET

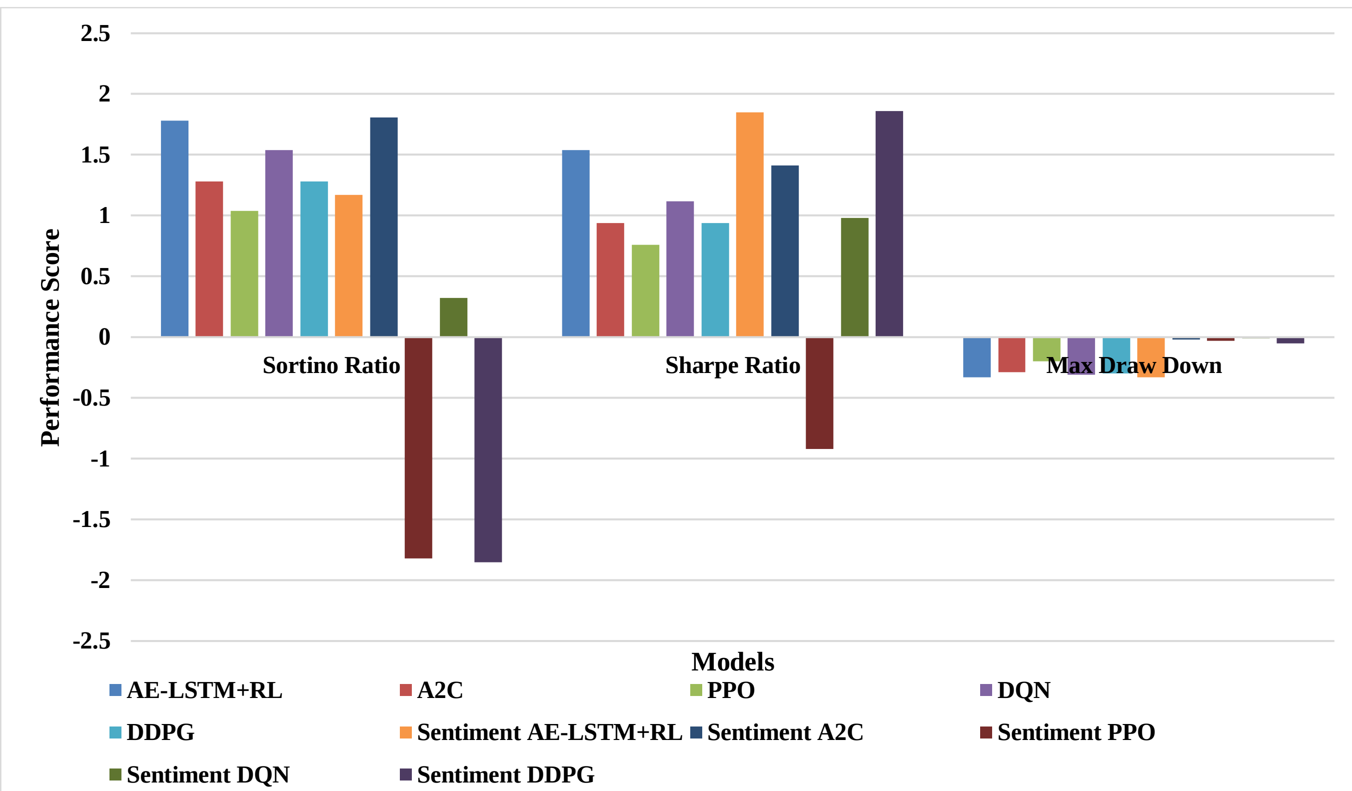
Data set	DJIA-STOCK MARKET					DJIA+TWITTER STOCK MARKET					S&P 500 STOCK MARKET					S&P 500 TWITTER STOCK MARKET				
Model	AE-LSTM+RL	A2C	PPO	DQN	DDPG	Sentiment AE-LSTM+RL	Sentiment A2C	Sentiment PPO	Sentiment DQN	Sentiment DDPG	AE-LSTM+RL	A2C	PPO	DQN	DDPG	Sentiment AE-LSTM+RL	Sentiment A2C	Sentiment PPO	Sentiment DQN	Sentiment DDPG
Sortino Ratio	1.04	1.09	0.64	-0.95	1.08	1.53	-0.92	1.16	-1.54	1.93	1.78	1.28	1.04	1.54	1.28	1.17	1.81	-1.82	0.32	-1.85
Sharpe Ratio	1.75	0.74	0.47	1.02	0.93	1.38	0.78	1.16	0.57	1.54	1.54	0.94	0.76	1.12	0.94	1.85	1.41	-0.92	0.98	1.86
Max Draw Down	-0.37	-0.39	-0.32	-0.3	-0.39	-0.21	-0.01	-0.04	-0.29	-0.4	-0.33	-0.29	-0.2	-0.31	-0.3	-0.33	-0.02	-0.03	-0.01	-0.05
Annual Return (%)	14	13	16	14	13	17	23	11	24	18	15	17	9	15	12	17	18	13	15	19
Cumulative Return (%)	72	68	35	56	64	62	25	18	20	50	78	81	56	42	35	70	32	52	20	46
Annual Volatility (%)	19	20	15	11	18	18	15	17	15	13	19	16	12	18	16	26	16	18	11	21



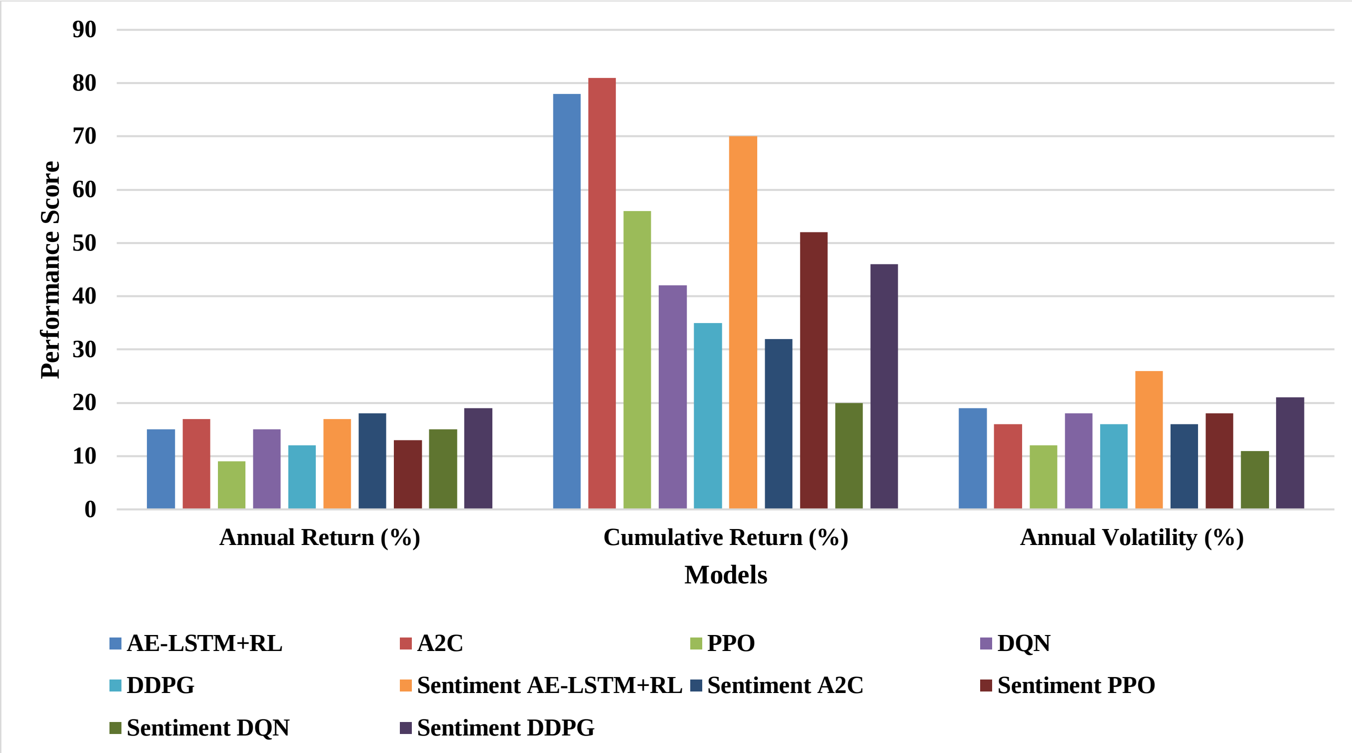
(a) Comparison of Performance Ratios of DJIA Stock Data and DJIAStock+Twitter Data Analysis



(b) Comparison of ROI measures of DJIA Stock Data and DJIAStock+Twitter Data Analysis



(c) Comparison of Performance Ratios of S&P 500 Stock Data and S&P 500 Stock+Twitter Data Analysis



(d) Comparison of RIO measures of S&P 500 Stock Data and S&P 500 Stock+Twitter Data Analysis
Figure 9: Performance Comparison Charts

Figure 9 displays the comparative performance of the algorithms in various aspects. Figure 9. shows the comparison of performance ratios of the proposed AE-LSTM+RL, A2C, PPO, DQN, and DDPG for DJIA stock data with and without Twitter data. Figure 9. b shows the comparison of ROI measures of the proposed AE-LSTM+RL, A2C, PPO, DQN, and DDPG for DJIA stock data with and without Twitter data. Figure 9. c shows the comparison of performance ratios of the proposed AE-LSTM+RL, A2C, PPO, DQN, and DDPG for S&P 500 stock data with and without Twitter data. Figure 9.d shows the comparison of ROI measures of the proposed AE-LSTM+RL, A2C, PPO, DQN, and DDPG for S&P 500 stock data with and without Twitter data. The results prove that the proposed AE-LSTM+RL works better compared to the existing methods in terms of both performance ratios and ROI measures for both datasets. It is noteworthy to mention that when the Twitter data is combined, the performance of the prediction of stock data is furthermore improved.

VI. CONCLUSION

In this paper, the S&P 500 stock index, Dow Jones, and tweets, normal trading conditions are investigated and further extending the study to evaluate their ability to simulate under distressed market conditions caused during the pandemic. The proposed system is a sentiment-based deep reinforcement learning method. Asper the obtained results, it is evident that AE-LSTM can detect the regular price accurately and the outcome from this study illustrates the flourishing adaptation of enhanced DLR algorithms to automate stock trading to increase profitability. The agent who is trained using the proposed method can take trading decisions more efficiently and can improve the results of portfolio allocation. From the results, it is evident that sentiment analysis impacts the automated trading system.

REFERENCES

[1] Kartika MaulidaHindrayani; Tresna Maulana Fahrudin; R. PrismahardiAji; Eristya Maya Safitri, Indonesian Stock Price Prediction including Covid19 Era Using Decision Tree Regression, 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), (2020).

[2] JinlongRuan, Wei Wu, Jiebo Luo, Stock Price Prediction Under Anomalous Circumstances, IEEE International Conference on Big Data (Big Data), (2020).

[3] Christopher Chou; Junho Park; Eric Chou, Predicting Stock Closing Price After COVID-19 Based on Sentiment Analysis and LSTM, IEEE, (2021).

[4] Jae Won idee, STOCK PRICE PREDICTION USING REINFORCEMENT LEARNING, IEEE, (2001).

[5] Kunal Pahwa; Neha Agarwal, Stock Market Analysis using Supervised Machine Learning, International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon), (2019).

[6] ZaharaddeenKaramiLawal; HayatiYassin; Rufai Yusuf Zakari, Stock Market Prediction using Supervised Machine Learning Techniques: An Overview, IEEE Asia-Pacific Conference on Computer Science and Data Engineering (CSDE), (2020).

[7] Jae Won Lee, Stock price prediction using reinforcement learning, IEEE International Symposium on Industrial Electronics Proceedings (Cat. No.01TH8570), (2001).

[8] V François-Lavet, P Henderson, R Islam. An introduction through deep reinforcement learningarXiv preprint arXi, (2018). - arxiv.org[PDF] arxiv.org.(

[9] PrahladKoratamaddi, Karan Wadhwani, Mridul Gupta a, Sriram G. Sanjeevi, Market sentiment-aware deep reinforcement learning approach for stock portfolio allocation, Engineering Science and Technology, an International Journal, (2021).

[10] Li, Y., Ni, P., & Chang, V. Application of deep reinforcement learning in stock trading strategies and stock forecasting. Computing, (2019) (1-18).

[11] Carta, S., Corriga, A., Ferreira, A., Podda, A. S., &Recupero, D. R. A multi-layer and multi-ensemble stock trader using deep learning and deep reinforcement learning., Applied Intelligence, 51(2) (2021) 889-905.

[12] Rundo, F. Deep LSTM with reinforcement learning layer for financial trend prediction in FX high-frequency trading systems. Applied Sciences, 9 (20) (2019) 4460.

[13] Long, W., Lu, Z., & Cui, L. Deep learning-based feature engineering for stock price movement prediction. Knowledge-Based Systems, 164 (2019) 163-173).

[14] Dang, Q. V. Reinforcement learning in stock trading. In International Conference on Computer Science, Applied Mathematics and Applications (2019) 311-322. Springer, Cham.

[15] Jia, W. U., Chen, W. A. N. G., Xiong, L., &Hongyong, S. U. N. Quantitative trading on the stock market based on deep reinforcement learning. In 2019 International Joint Conference on Neural Networks (IJCNN), IEEE, (2019) (1-8).

[16] Huotari, T., Savolainen, J., &Collan, M. Deep reinforcement learning agent for S&P 500 stock selection. Axioms, 9(4) (2020) 130.

[17] M. Baker, J. Wurgler Investor sentiment in the stock market J. Econ. Perspect., 21 (2007) 129-152. 10.1257/jep.21.2.12.

[18] 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.

[19] Wei Bao, Jun Yue, Yulei Rao, A deep learning framework for financial time series using stacked autoencoders and long short-term memory, (2017).

[20] 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, (2004).

[21] J Patel, S Shah, P Thakkar, K Kotecha, Predicting stock market index using a fusion of machine learning techniques, Expert Systems with Applications, (2015). Elsevier

[22] V Mnih, AP Badia, M Mirza, Graves Asynchronous methods for deep reinforcement learning The 33rd International Conference on Machine Learning, 02 (2016).[PDF] mlr.press

[23] J. Bollen, H. Mao, X. Zeng, Twitter mood predicts the stock market, J. Comput. Sci., 1-8. 10.1016/j.jocs.2010.12.007,2011.

[24] Mehrnaz Faraz, Hamid Khaloozadeh, Milad Abbasi, Stock Market Prediction-by-Prediction Based on Autoencoder Long Short-Term Memory Networks, Conference Paper • December, (2020). DOI: 10.1109/ICEE50131.2020.9261055.

[25] M Van Otterlo, M Wiering Reinforcement learning & Markov decision processes Reinforcement learning, (2012) – Springer[PDF] rug.nl

[26] S Carta, Ferreira, AS Podda, DR Recupero, Multi-DQN: An ensemble of Deep Q-learning agents for stock market forecasting unicamp. br

[27] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov, Proximal policy optimization algorithms, arXiv:1707.06347, 07 (2017).

[28] T Théate, D Ernst An application of deep reinforcement learning through algorithmic trading[PDF] arxiv.org

[29] B. Yang, Z.-J. Gong, and W. Yang, Stock market index prediction using deep neural network ensemble, in 2017 36th Chinese Control Conference (CCC). IEEE, (2017) 3882–3887.

[30] 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.