CSCE 790-002: Deep Reinforcement Learning and Search Progress Report

Project Title: Stock Trading Automation with Deep Reinforcement Learning
Team Members: Hong Yung Yip
Date: 12/11/20

Abstract

Predicting the stock behavior based on prior data is still an open problem due to the adaptive nature of the environment. Current state-of-the-art (SOTA) in automated stock trading and price prediction is the Deep Reinforcement Learning (DRL) approach, specifically the Deep Deterministic Policy Gradient (DDPG) which outperformed the Dow Jones Industrial Average annualized return by 16.40%. Nonetheless, the approach only considers the historical prices, albeit one of the main factors driving a company stock prices is public sentiments and past stock performance (movements). Therefore, this project aims to investigate additional trading indicators: news sentiment analysis and temporal price movements to improve the SOTA on three company stocks: Apple, Boeing, and Tesla, due to good price volatilities based on recent (2-month) news. The news sentiment is analyzed by the NLTK Vader (a dictionary-based approach to score sentiment based on positive and negative lexicons), while the historic price movement is modelled by the Long-Short Term Memory (LSTM) approach. The additional indicators produced by the respective models (sentiment Vader and LSTM) are then represented as additional states to the DRL models: Deep Q-Network (DQN) and DDPG. DQN with onestep forecast yields an approximately 1.9 times increase compared to the initial capital balance in a simulated MDP stock environment, which demonstrates the preliminary potential use of additional trading signals to improve the DRL agent's learning. Future work includes the use of bigger datasets and a more vigorous approach to news data collection to further improve on the current DRL performance.

Introduction

Learning to predict market performance based on historical data remains an open challenge despite the advent of Big Data and machine/ deep learning technologies. The increasing complexity and dynamical property of the stock markets as well as public (crowd) sentiments are some of the partially observable factors that contribute to the sea of uncertainty when predicting future stocks performance [1]. Although there has been adoption of computer-aided stock trading [2], increasing research on designing profitable strategies [3-5], algorithmic modelling of the stock market, the true state as well as the transition dynamics of the stock environment remain unsolved.

Stock trading is a constant and iterative process of testing new ideas, receiving market feedback (profit/ loss), and from these experiences, learning to improve and optimize trading strategies (policies) over time [4]. A profitable stock trading strategy is one that optimizes allocation of capital and maximizes return. This trial-and-error approach to decision making coupled with the partially observable Markov environment make this an interesting and well-suited problem for deep reinforcement learning (DRL).

Related Work

Studies to accurately predict the stock market remains an on-going research due to its volatile and non-linear nature. Despite existing studies on using supervised (eg. ANN, RNN, and LSTM) and unsupervised approaches (eg. K-means) [6], results were still far from satisfactory mainly because relying on historical prices (open, high, low, and close) and trading indicators (buy and sell volume, market trends, and moving averages) alone were not enough to be an indicative of future market behavior. On the contrary, the use of DRL techniques are on the rise [2-4] due to how the stock market environment can be modelled as a partially observable Markov decision process (MDP) (stock prices) with agent learning to optimize allocation of capital (buy, sell, hold) and maximize expected return of investment (policy) with respect to the overall portfolio value (reward function).

Recent state-of-the-art (SOTA) on automated stock trading with Deep Deterministic Policy Gradient (DDPG) [3] and Ensemble Strategy [4] has shown promising results on outperforming the Dow Jones Industrial Average and in accumulated portfolio return. However, these approaches only consider the historical daily prices as the states. At the granular level, one of the main factors driving a company stock prices is investors' demand. Demand can be influenced by market's (investor and consumer) expectations and emotions [1]. The voluminous and ubiquity of data (news data) today has enabled individuals at any scale to make analytical investment decisions. Motivated by these challenges, this project attempts to explore and differentiate itself by proposing and including the additional trading indicators into a Deep Q-Network (DQN): (1) news sentiment metadata, and (2) one-step stock prediction into the existing states [1] (refer Figure 1). The rationale is to train a trading agent that will trade not only by historical prices, but also take in account the news sentiment signals [6] when deciding to buy, sell or hold a particular stock. The DQN network with additional indicators is then compared with the baseline DQN and the current state-of-the-art, DDPG [3].

Approach

The main objective of this project is to investigate whether additional trading signals: news sentiment and one-step stock forecast will improve the current SOTA. The contributions of this project are four-fold:

1. **Baseline:** Reproduce (partially) the current SOTA, DDPG [3], and compare it with DQN. The main difference between DDPG and DQN lies in the Q-(action) values that are produced by the models, DDPG being continuous (hence the possibility of fractional shares trading) and DQN being discrete.
2. Determine the adaptive **effects of news sentiment polarity** (represented as additional states) as discounting factors in improving the models' performance.
3. Determine the **effects of onestep stock price forecast** by modeling the temporal nature (additional states) of the stock environment in encouraging the agents' learning.
4. Evaluate the effects of (1), (2), and (3) at a holistic level.

The scope of the project is further divided into different components: (a) news and financial data collection and preprocessing, (b) news sentiment analysis, (c) one and multistep stock price forecast, and (d) MDP formulation and DRL models training and evaluation.

(a) News and Financial Data Collection

News and financial data for the period (10/9/20 - 11/29/20) has been collected for the following stocks: (1) AAPL (Apple), (2) BA (Boeing), and (3) TSLA (Tesla) from NewsAPI (news) and AlphaVantage (daily price) respectively[1] (Table 1). The three stocks above experienced volatility in the past month due to major news events such as Apple released iPhone 12, Presidential elections, and COVID-19 which makes an interesting task for a DRL agent to optimize the "best" trading strategy. The daily stock prices consist of Open, High, Low, Close, and Volume. The **daily *open* price** is used as it accounts for price movement based on the news for the entire (previous) day including after-market trading hours.

Table 1: Statistics of the news and financial data of each company respectively

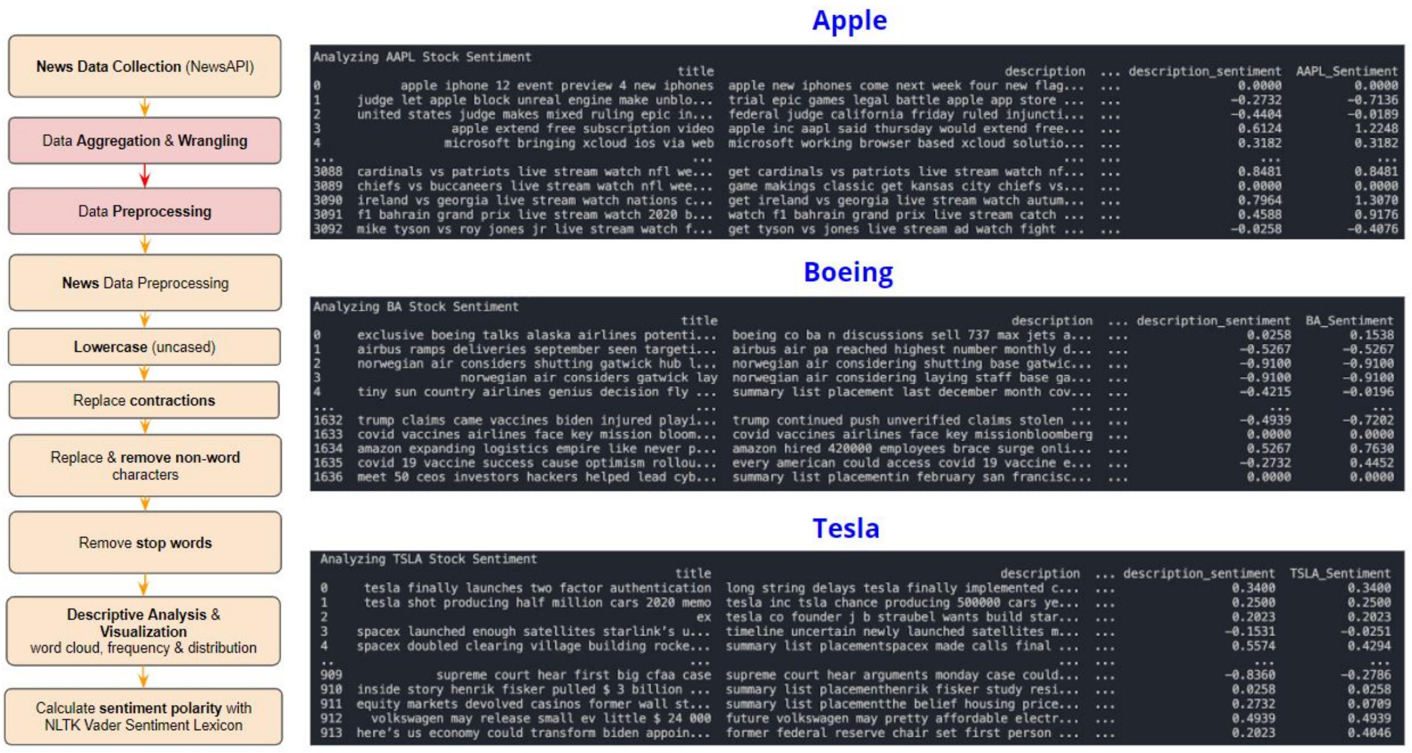
News and Financial Data for the period (10/9/20 - 11/29/20)							
Sources: bloomberg, cnn, cnbc, business-insider, financial-post, fortune, hacker-news, time, wired, techcrunch, techradar, the-verge, the-washington-times, engadget, mashable, ars-technica, reuters, google-news, newsweek, next-big-future, politico							
Stock	**Keywords used to collect news data**	**Number of news collected**	**Size of vocabulary**	**Average news title length**	**Average news description length**	**Lowest (Open) Price**	**Highest (Open) Price**
AAPL (Apple)	apple, aapl, iphone, 5G	1499	7758	28	40	109.11	125.27
BA (Boeing)	boeing, airline, airlines, aviation	828	5347	26	36	145.75	179.0
TSLA (Tesla)	tesla, tsla, elon, musk	477	4028	27	39	394.0	454.44

[1] The data are only collected up to a month period due to the limitation imposed by NewsAPI on developer account.

Example News Data (Title and Description)
News title: Apple just announced its 5th stock split in history[2].
News description: Apple will split its stock for the fifth time in its history, the iPhone maker said on Thursday after reporting strong earnings that beat analysts' estimates. Apple's stock price will be quartered, from about $400 now to about $100 when the split happens.

News Data Preprocessing

The collected news text data (both title and description) are subjected to a series of standard NLP preprocessing pipeline (Figure 1) prior to sentiment analysis. The news text data are (1) lowercase, (2) contractions replaced, (3) non-word characters (symbols, punctuations, white spaces, and tabs) replaced, (4) and stop-words removed using a combination of both rule-based approach and the Natural Language Toolkit (NLTK) library. Then, a series of descriptive analysis such as word cloud, frequency, and distribution are performed to visualize the news content.


News Data Collection (NewsAPI)
Data Aggregation & Wrangling
Data Preprocessing
News Data Preprocessing
Lowercase (uncased)
Replace contractions
Replace & remove non-word characters
Remove stop words
Descriptive Analysis & Visualization word cloud, frequency & distribution
Calculate sentiment polarity with NLTK Vader Sentiment Lexicon
Apple
Boeing
Tesla

```
Analyzing AAPL Stock Sentiment
                                            title                                         description  ... description_sentiment  AAPL_Sentiment
0          apple iphone 12 event preview 4 new iphones  apple new iphones come next week four new flag...  ...                0.0000          0.0000
1     judge let apple block unreal engine make unblo...  trial epic games legal battle apple app store ...  ...               -0.2732         -0.7136
2     united states judge makes mixed ruling epic in...  federal judge california friday ruled injuncti...  ...               -0.4404         -0.0189
3              apple extend free subscription video  apple inc aapl said thursday would extend free...  ...                0.6124          1.2248
4             microsoft bringing xcloud ios via web  microsoft working browser based xcloud solutio...  ...                0.3182          0.3182
...                                             ...                                                ...  ...                   ...             ...
3088  cardinals vs patriots live stream watch nfl we...  get cardinals vs patriots live stream watch nf...  ...                0.8481          0.8481
3089  chiefs vs buccaneers live stream watch nfl wee...  game makings classic get kansas city chiefs vs...  ...                0.0000          0.0000
3090  ireland vs georgia live stream watch nations c...  get ireland vs georgia live stream watch autum...  ...                0.7964          1.3070
3091  f1 bahrain grand prix live stream watch 2020 b...  watch f1 bahrain grand prix live stream catch ...  ...                0.4588          0.9176
3092  mike tyson vs roy jones jr live stream watch f...  get tyson vs jones live stream ad watch fight ...  ...               -0.0258         -0.4076
```

```
Analyzing BA Stock Sentiment
                                             title                                         description  ... description_sentiment  BA_Sentiment
0     exclusive boeing talks alaska airlines potenti...  boeing co ba n discussions sell 737 max jets a...  ...                0.0258        0.1538
1     airbus ramps deliveries september seen targeti...  airbus air pa reached highest number monthly d...  ...               -0.5267       -0.5267
2     norwegian air considers shutting gatwick hub l...  norwegian air considering shutting base gatwic...  ...               -0.9100       -0.9100
3                norwegian air considers gatwick lay  norwegian air considering laying staff base ga...  ...               -0.9100       -0.9100
4     tiny sun country airlines genius decision fly ...  summary list placement last december month cov...  ...               -0.4215       -0.0196
...                                              ...                                                ...  ...                   ...           ...
1632  trump claims came vaccines biden injured playi...  trump continued push unverified claims stolen ...  ...               -0.4939       -0.7202
1633  covid vaccines airlines face key mission bloom...  covid vaccines airlines face key missionbloomberg  ...                0.0000        0.0000
1634  amazon expanding logistics empire like never p...  amazon hired 420000 employees brace surge onli...  ...                0.5267        0.7630
1635  covid 19 vaccine success cause optimism rollou...  every american could access covid 19 vaccine e...  ...               -0.2732        0.4452
1636  meet 50 ceos investors hackers helped lead cyb...  summary list placementin february san francisc...  ...                0.0000        0.0000
```

```
Analyzing TSLA Stock Sentiment
                                            title                                         description  ... description_sentiment  TSLA_Sentiment
0      tesla finally launches two factor authentication  long string delays tesla finally implemented c...  ...                0.3400          0.3400
1      tesla shot producing half million cars 2020 memo  tesla inc tsla chance producing 500000 cars ye...  ...                0.2500          0.2500
2                                                    ex  tesla co founder j b straubel wants build star...  ...                0.2023          0.2023
3    spacex launched enough satellites starlink’s u...  timeline uncertain newly launched satellites m...  ...               -0.1531         -0.0251
4    spacex doubled clearing village building rocke...  summary list placementspacex made calls final ...  ...                0.5574          0.4294
..                                                ...                                                ...  ...                   ...             ...
909            supreme court hear first big cfaa case  supreme court hear arguments monday case could...  ...               -0.8360         -0.2786
910  inside story henrik fisker pulled $ 3 billion ...  summary list placementhenrik fisker study resi...  ...                0.0258          0.0258
911  equity markets devolved casinos former wall st...  summary list placementthe belief housing price...  ...                0.2732          0.0709
912    volkswagen may release small ev little $ 24 000  future volkswagen may pretty affordable electr...  ...                0.4939          0.4939
913  here’s us economy could transform biden appoin...  former federal reserve chair set first person ...  ...                0.2023          0.4046
```

Figure 1: News data preprocessing and sample data

(b) News Sentiment Analysis

The **NLTK Vader** (Valence Aware Dictionary and sEntiment Reasoner) Sentiment Analysis[3] library is used for automatic sentiment detection of news data (titles and descriptions). It is a type of (pre-trained dictionary-based) sentiment analysis that is based on lexicons of sentiment-related words, which takes on two forms: (1) polarity-based, where pieces of texts are classified as either positive or negative, and (2) valence-based, where the intensity of the sentiment is taken into account. In other words, each word in the lexicon is rated as how positive or negative. It analyzes a piece of text by (tokenizing and) checking the number of words that are present in its lexicon and produces four sentiment metrics (positive, neutral, negative, and compound). In this project, the input is the news data (title and description respectively) and the output is the **compound** score (the sum of news title and description sentiment scores aggregated on a daily basis), which have been standardised to range between -1 and 1, is used as the sentiment state inputs.

[2]https://markets.businessinsider.com/news/stocks/apple-stock-split-price-what-that-means-how-many-shares-2020-7-1029455711#
[3] https://t-redactyl.io/blog/2017/04/using-vader-to-handle-sentiment-analysis-with-social-media-text.html

(c) Onestep Stock Price Forecast

The **Long-Short Term Memory (LSTM)** network is used as it is a type of recurrent model that excels at modeling sequential information due to the presence of memory cells to store and "remember" data read over time. In addition, it is capable of handling long term dependencies and minimizing the inherent vanishing gradient problem of vanilla recurrent networks (RNN) with the introduction of "gates" (input, output and forget gates) to control the flow of and retain information better through time [9]. The onestep stock price forecast takes into input a sliding window of 10, in other words, the price history of the last 10 days to predict the price (numerical value) of the following day (output). The LSTM (5 units) is optimized with the Adam optimizer with learning rate of 0.1 and the mean-squared error (MSE) as the loss function. There are three different LSTM networks in total, each trained on a specific stock respectively as LSTM trained on AAPL will not be able to predict TSLA price. Overall, all three networks achieved, on average, 90% training and validation accuracy (Figure 2).

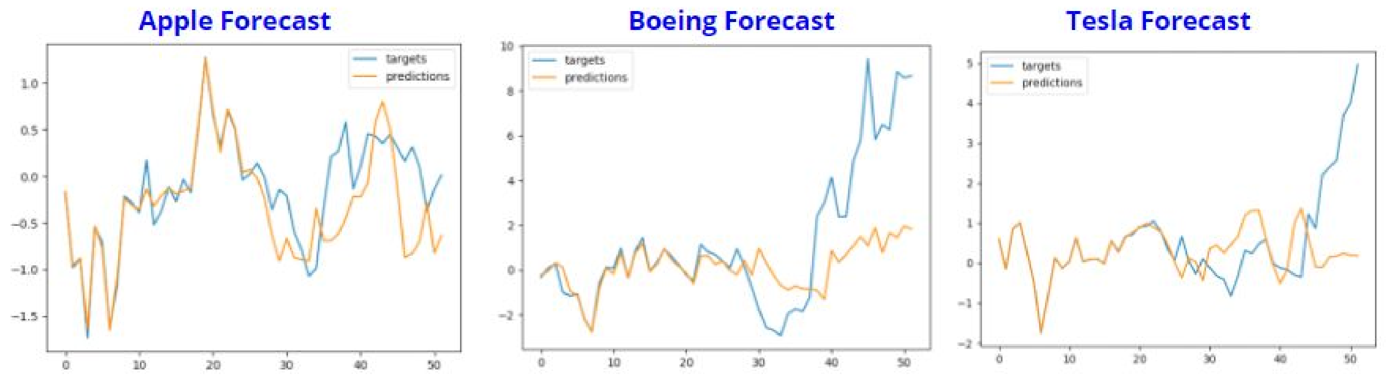


Figure 2: Onestep stock price forecast with LSTM

(d) MDP Formulation and DRL Models

A simulated stock trading environment (Figure 3) with the following Markov Decision Process (MDP) has been implemented:

- **State, s** in the (baseline) form includes the information of the stock prices $p \in R^D$, the amount of holdings of stocks $h \in Z^D$, and the remaining balance $b \in R$, where D is the number of stocks that we consider in the market and Z denotes non-negative integer numbers. Example representation as follows: [# of AAPL shares, # of TSLA shares, # of JNJ shares, current price of AAPL, current price of BA, current price of TSLA, current capital balance].
- **Action, a**, for each stock include selling, buying, and holding. D stocks with 3 action permutations translates to 27 possible actions.
- **Reward r(s,a,s')** is the change of the portfolio value when action a is taken at state s and arriving at the new state s'. The portfolio value is the sum of the equities in all held stocks.
- **Policy π(s)** is the trading strategy of stocks at state s in the following tuple (buy, buy, sell) corresponding to buy AAPL, buy BA, and sell TSLA.
- **Action-value function Qπ(s, a)** is the expected reward achieved by action a at state s following policy π.

The expected reward of taking action, a_t is calculated by taking the expectation of the rewards $r(s_t, a_t, s_{t+1})$ plus the expected reward in the next state, s_{t+1}, according to the Bellman Equation.

$$Q_\pi(s_t, a_t) = \mathbb{E}_{s_{t+1}}[r(s_t, a_t, s_{t+1}) + \gamma \mathbb{E}_{a_{t+1} \sim \pi(s_{t+1})}[Q_\pi(s_{t+1}, a_{t+1})]].$$

(e) Portfolio Value Maximization as Trading Goal

The goal of this project is to maximize the portfolio value return at a target time (t) in the future. Due to the Markov property of the DRL (both DQN and DDPG) models, the trading objective can be formulated to optimize the policy that maximizes the function **Qπ(s_t, a_t).**

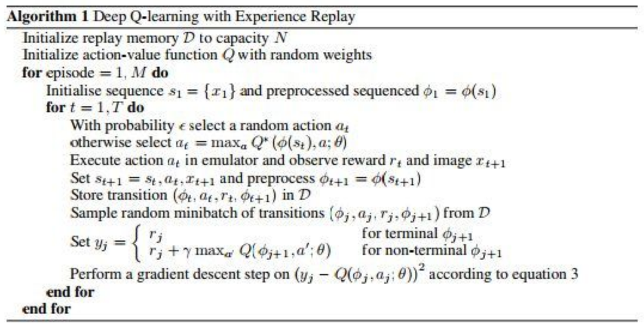
(f) Deep Reinforcement Learning Approach

For brevity, this report will not discuss the technical depths of each DRL model, but rather a holistic view of each approach. **Q-learning** is an off-policy method that uses epsilon-greedy policy to explore-exploit and maximize the Q(s_{t+1}, a_{t+1}) for state $s_{s+1.}$ However, as the number of states and actions in the environment increases, Q-learning may not be feasible due to the curse of dimensionality.

$$Q_\pi(s_t, a_t) = \mathbb{E}_{s_{t+1}}[r(s_t, a_t, s_{t+1}) + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1})].$$

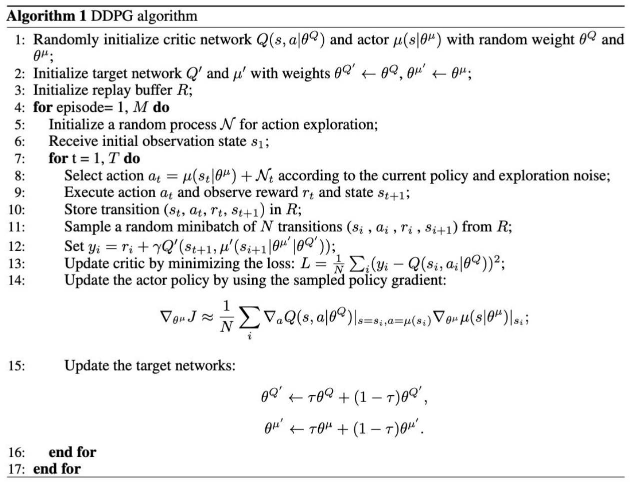
On the contrary, Deep Q-learning or **DQN** (with experience replay) (Figure 3) uses neural networks as the function approximators which present itself as the viable approach for approximating larger environments. Nonetheless, according to [3], DQN will eventually become intractable as the action spaces grow exponentially. Hence, DDPG (Figure 4) is proposed to deterministically map states to actions to address the challenge.

DDPG (Figure 4) maintains two networks: an actor and a critic, where the actor network maps states to actions, and the critic network outputs the action value under state s. The critic network is updated by minimizing the expected difference between the outputs from the target critic network and the "main" critic network. Both DQN and DDPG employ an experience replay buffer to store transitions to minimize the correlation between experience samples, which is important as we do not want the DRL models to copy the trend based on preceding trends. Both have target networks to provide temporal difference updates and improve convergence (as we do not want the main network to be in a tail-biting scenario).



```
Algorithm 1 Deep Q-learning with Experience Replay
Initialize replay memory $\mathcal{D}$ to capacity $N$
Initialize action-value function $Q$ with random weights
for episode = 1, $M$ do
    Initialise sequence $s_1 = \{x_1\}$ and preprocessed sequenced $\phi_1 = \phi(s_1)$
    for $t = 1, T$ do
        With probability $\epsilon$ select a random action $a_t$
        otherwise select $a_t = \max_a Q^*(\phi(s_t), a; \theta)$
        Execute action $a_t$ in emulator and observe reward $r_t$ and image $x_{t+1}$
        Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$
        Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in $\mathcal{D}$
        Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from $\mathcal{D}$
        Set $y_j = \begin{cases} r_j & \text{for terminal } \phi_{j+1} \\ r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta) & \text{for non-terminal } \phi_{j+1} \end{cases}$
        Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ according to equation 3
    end for
end for
```

Figure 3: DQN Algorithm [10]



```
Algorithm 1 DDPG algorithm
1: Randomly initialize critic network $Q(s, a|\theta^Q)$ and actor $\mu(s|\theta^\mu)$ with random weight $\theta^Q$ and $\theta^\mu$;
2: Initialize target network $Q'$ and $\mu'$ with weights $\theta^{Q'} \leftarrow \theta^Q$, $\theta^{\mu'} \leftarrow \theta^\mu$;
3: Initialize replay buffer $R$;
4: for episode= 1, $M$ do
5:     Initialize a random process $\mathcal{N}$ for action exploration;
6:     Receive initial observation state $s_1$;
7:     for t = 1, $T$ do
8:         Select action $a_t = \mu(s_t|\theta^\mu) + \mathcal{N}_t$ according to the current policy and exploration noise;
9:         Execute action $a_t$ and observe reward $r_t$ and state $s_{t+1}$;
10:        Store transition $(s_t, a_t, r_t, s_{t+1})$ in $R$;
11:        Sample a random minibatch of $N$ transitions $(s_i, a_i, r_i, s_{i+1})$ from $R$;
12:        Set $y_i = r_i + \gamma Q'(s_{t+1}, \mu'(s_{i+1}|\theta^{\mu'}|\theta^{Q'}))$;
13:        Update critic by minimizing the loss: $L = \frac{1}{N}\sum_i (y_i - Q(s_i, a_i|\theta^Q))^2$;
14:        Update the actor policy by using the sampled policy gradient:
               $\nabla_{\theta^\mu} J \approx \frac{1}{N} \sum_i \nabla_a Q(s, a|\theta^Q)|_{s=s_i, a=\mu(s_i)} \nabla_{\theta^\mu} \mu(s|\theta^\mu)|_{s_i};$
15:        Update the target networks:
               $\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'},$
               $\theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}.$
16:    end for
17: end for
```

Figure 4: DDPG Algorithm [3]

(g) Experiments Setup

A total of eight experimental and ablation setups are performed to investigate the role of each trading signals:

1. Baseline DQN
2. DQN with onestep forecast
3. DQN with news sentiment
4. DQN with onestep forecast and news sentiment
5. Baseline DDPG
6. DDPG with onestep forecast
7. DDPG with news sentiment
8. DDPG with onestep forecast and news sentiment

The overall architecture with respective state representations, models, and outputs are illustrated in Figure 3.

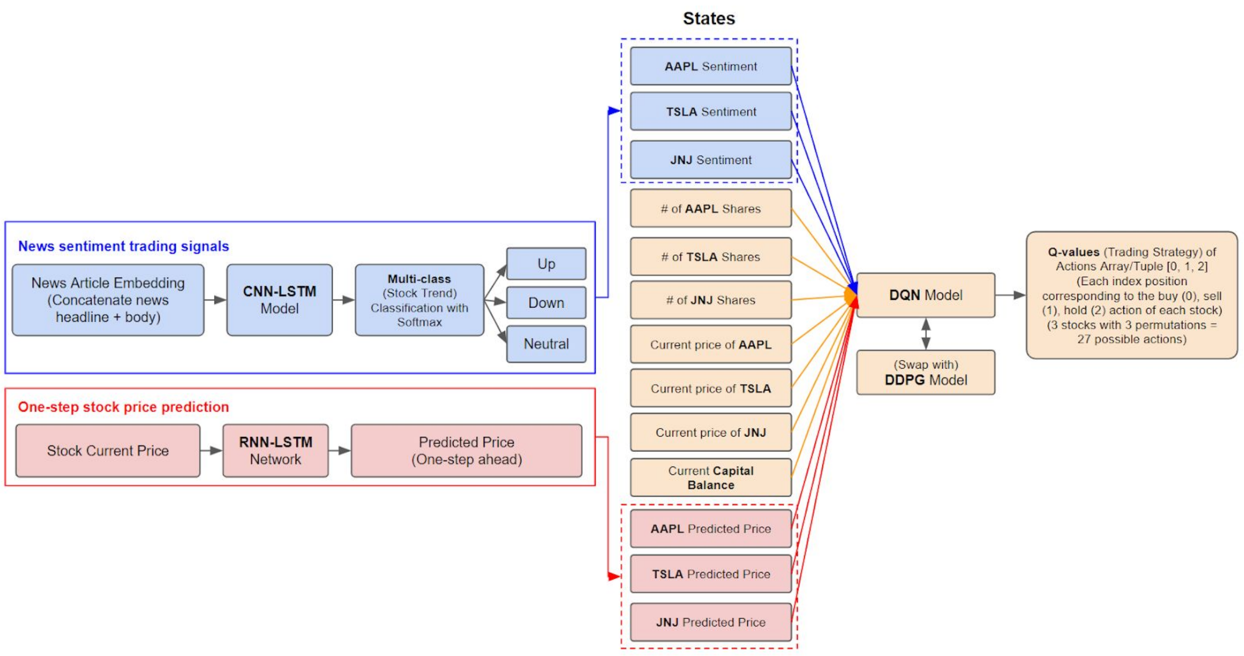

States
AAPL Sentiment
TSLA Sentiment
JNJ Sentiment
# of AAPL Shares
# of TSLA Shares
# of JNJ Shares
Current price of AAPL
Current price of TSLA
Current price of JNJ
Current Capital Balance
AAPL Predicted Price
TSLA Predicted Price
JNJ Predicted Price
News sentiment trading signals
News Article Embedding (Concatenate news headline + body)
CNN-LSTM Model
Multi-class (Stock Trend) Classification with Softmax
Up
Down
Neutral
One-step stock price prediction
Stock Current Price
RNN-LSTM Network
Predicted Price (One-step ahead)
DQN Model
(Swap with) DDPG Model
Q-values (Trading Strategy) of Actions Array/Tuple [0, 1, 2] (Each index position corresponding to the buy (0), sell (1), hold (2) action of each stock) (3 stocks with 3 permutations = 27 possible actions)

Figure 3: The overall architecture (states representation, models, and outputs) for this project. Boxes highlighted in Orange are the baseline DQN model. Boxes highlighted in **Blue** are the additional news sentiment metadata trained using an external CNN-LSTM network and the output of the model (stock trend movement) will be the additional state inputs on top of the baseline states (in Orange). Boxes highlighted in **Red** are the one-step stock price that is trained with an external LSTM network and the output (predicted price) will be used as additional trading signals.

Experimental Results

The metric used to measure the models' performance is the cumulative portfolio return, which reflects the portfolio value at the end of the episodes (Table 2).

Table 2: Cumulative portfolio return (max. of all episodes) and return of investment (fold) of each model respectively

Models *(Epsilon-greedy policy of 0.6)*	**Initial Capital**	**Cumulative Portfolio Return** *(Max of all episodes)*	**Return of Investment (In Fold)**
DQN	10,000	14,846.29	~1.5x
DQN + Onestep Forecast	**10,000**	**19,500.61**	**~1.9x**
DQN + News Sentiment	10,000	17,710.09	~1.7x
DQN + Onestep Forecast + News Sentiment	10,000	18,112.42	~1.8x
DDPG	10,000	14.185.75	~1.4x
DDPG + Onestep Forecast	10,000	13,185.75	~1.3x
DDPG + News Sentiment	10,000	12,560.93	~1.25x
DDPG + Onestep Forecast + News Sentiment	10,000	13,061.52	~1.3x

The results demonstrate that DQN with onestep forecast performs the best, with a return of approximately 1.9 times initial capital as LSTM models are able to capture the stock prices and trends (Figure 2) with over 90% accuracy. Nonetheless, the results **differ from the current SOTA literature** that claimed DDPG [3] will emerge as the best performing model and some of the plausible explanations are:

- Due to the limitation of publicly available **news data** collection APIs, the two month period of data collection (10/9/20 - 11/29/20) is too short for any significant stock trends movement.
- The labeling schema (difference window) for trends movement is restricted due to the short datasets period.
- The random walk nature of the stock movements (in these datasets)
- The keywords used to search the news data play a crucial role in the quality of datasets being collected. As noted in the BA datasets, the keywords used (boeing, airline, airlines, aviation) are simply too broad. Upon sampling a subset of news data, there are many “noisy” news data such as advertisements, and paid articles, hence producing mixed sentiment signals.
- DDPG requires a relatively substantial amount of data for it to “learn” the environment effectively. Existing SOTA literature [3] uses data from 1/1/2009 to 1/1/2016 across Dow Jones 30 stocks.

Conclusion

This project aims to reproduce existing research and investigate the potential improvements by incorporating additional feature components (NLP and temporal) with DRL to predict the stock markets and perform automated stock trading. The best-performing model, DQN with onestep forecast trained on historical stocks (AAPL, BA, and TSLA) prices in a simulated MDP environment yields an approximately 1.9 times increase compared to the initial capital balance, which reinforces the hypothesis that (additional) quality feature has a positive impact of the DRL agents. With the challenges mentioned in the “Experimental Results” section, future work includes the use of bigger datasets with keyword engineering for data collection and a more rigorous data scrutinization (filtering) on irrelevant news to remove the quality of news data.

References

[1] Wu, X., Chen, H., Wang, J., Troiano, L., Loia, V., & Fujita, H. (2020). Adaptive Stock Trading Strategies with Deep Reinforcement Learning Methods. Information Sciences.

[2] Carta, S., Corriga, A., Ferreira, A., Podda, A. S., & Recupero, D. R. (2020). A multi-layer and multi-ensemble stock trader using deep learning and deep reinforcement learning. Applied Intelligence, 1-17.

[3] Xiong, Z., Liu, X. Y., Zhong, S., Yang, H., & Walid, A. (2018). Practical deep reinforcement learning approach for stock trading. arXiv preprint arXiv:1811.07522.

[4] Yang, H., Liu, X. Y., Zhong, S., & Walid, A. (2020). Deep Reinforcement Learning for Automated Stock Trading: An Ensemble Strategy. Available at SSRN.

[5] Vijh, M., Chandola, D., Tikkiwal, V. A., & Kumar, A. (2020). Stock Closing Price Prediction using Machine Learning Techniques. Procedia Computer Science, 167, 599-606.

[6] Powell, N., Foo, S. Y., & Weatherspoon, M. (2008, March). Supervised and unsupervised methods for stock trend forecasting. In 2008 40th Southeastern Symposium on System Theory (SSST) (pp. 203-205). IEEE.

[7] Moghar, A., & Hamiche, M. (2020). Stock Market Prediction Using LSTM Recurrent Neural Network. Procedia Computer Science, 170, 1168-1173.

[8] Bharathi, S., & Geetha, A. (2017). Sentiment analysis for effective stock market prediction. International Journal of Intelligent Engineering and Systems, 10(3), 146-154.

[9] Rychalska, B., Pakulska, K., Chodorowska, K., Walczak, W., Andruszkiewicz, P.: Samsung Poland NLP Team at SemEval-2016 Task 1: Necessity for diversity; combining recursive autoencoders,WordNet and ensemble methods to measure semantic similarity. In Proceedings of the 10th International.

[10] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602.