Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2022) 30: 1506 – 1524

doi:10.55730/1300-0632.3862

A novel deep reinforcement learning based stock price prediction using knowledge graph and community aware sentiments

Anıl Berk ALTUNER, Zeynep Hilal KİLİMCİ*
Department of Information Systems Engineering, Faculty of Technology, Kocaeli University, İzmit, Turkey

Received: 10.09.2021 • **Accepted/Published Online:** 15.04.2022 • **Final Version:** 31.05.2022

Abstract: Stock market prediction has been an important topic for investors, researchers, and analysts. Because it is affected by too many factors, stock market prediction is a difficult task to handle. In this study, we propose a novel method that is based on deep reinforcement learning methodologies for the prediction of stock prices using sentiments of community and knowledge graph. For this purpose, we firstly construct a social knowledge graph of users by analyzing relations between connections. After that, time series analysis of related stock and sentiment analysis is blended with deep reinforcement methodology. Turkish version of Bidirectional Encoder Representations from Transformers (BerTurk) is employed to analyze the sentiments of the users while deep Q-learning methodology is used for the deep reinforcement learning side of the proposed model to construct the deep Q-network. In order to demonstrate the effectiveness of the proposed model, Garanti Bank (GARAN), Akbank (AKBNK), Türkiye İş Bankası (ISCTR) stocks in Borsa İstanbul are used as a case study. Experiment results show that the proposed novel model achieves remarkable results for stock market prediction task.

Key words: Deep reinforcement learning, knowledge graphs, sentiment analysis, social graphs, stock prediction, q-learning

1. Introduction

Prediction of stock price index is seen as one of the most challenging applications of time series forecasting. There are many studies that address the issues of forecasting the stock price index in advanced financial markets and emerging markets such as Turkey stock exchange. Consistent forecasts made with stock price indices are important for the development of effective market trading strategies [1]. In this way, investors can protect against potential market risks and speculators. In addition, they can have the opportunity to make a profit by trading in the stock index [2]. The stock market is essentially dynamic, linear and nonparametric, complex and chaotic structure, stock market forecasting, financial time series forecasting process is seen as a challenging task [3]. In addition, the stock market is affected by many macroeconomic factors, such as political events, firm policies, general economic conditions, investors' expectations, choices of institutional investors, movements of other exchanges, and investor psychology [4]. Sentiment analysis of financial text has been an important and active research topic for analysts and investors in forecasting stock prices or directions. It has been observed that opinions may affect market dynamics [5].

Knowledge graph is a knowledge base that employs a graph-structured data model in order to consolidate data. Knowledge graphs are often used to store interlinked descriptions of entities such as objects, events,

*Correspondence: zeynep.kilimci@kocaeli.edu.tr

situations or abstract concepts with free-form semantics. The knowledge graph symbolizes a aggregation of interrelated specifications of entities (objects, events or concepts) and their relationships. Knowledge graphs ensure a platform for data consolidation, association, analytic, joining by putting data in context through semantic meta data. Knowledge graphs may employ ontology as a schema stage. By doing this, they consent logical deduction for revoking implicit knowledge rather than just letting queries demanding explicit knowledge. In this work, we use the knowledge graph when inferring texts/comments with semantic convergence about stocks. Furthermore, social graph is a diagram that illustrates interconnections among people, groups and organizations in a social network. The term is also used to describe an individual's social network [6]. The social graph shape appears as a series of network nodes connected by lines. Nodes on the graph represent an object, and paths between each object are called edges. Edge can be of more than one type, so the link between two objects can be associated. Instead of collecting random users' comments about related stocks, we also use social graphs to gather comments from followers of people with a large number of followers. In this way, we collect comments that contain the thoughts of people who are more relevant or knowledgeable about the related stock.

Sentiment analysis is a classification task that measures the emotional level in the discourse. The opinion can be subjective assessment of something based on personal experience or an aspect for particular issue. Sentiment analysis can be used on finding and extracting the opinionated data on a platform, define subject matter or determine its polarity. Although there are too many classification techniques in order to determine sentiment of opinion, there are very limited state-of-the-art studies on deep reinforcement learning based sentiment analysis. In this study, we propose a novel deep reinforcement learning based stock price prediction using knowledge graph and community aware sentiments.

Reinforcement learning (RL) is a one of the artificial intelligence techniques that an agent to learn in an interactive environment by trial and error using feedback from its own experiences. There is a situation where the agent is positioned according to the value obtained as a result of every action he makes. The results obtained from the movements can be called reward and punishment. RL offers reward mechanism and create policy for trading. Reinforced learning acquires a behavioral gain by learning this policy through trial-and-error method. Q-learning and SARSA are commonly employed algorithms in many artificial intelligence applications (Alpha Go Zero) and researches [7, 8]. Deep reinforcement learning is the combination of reinforcement learning and deep learning that is being able to solve a wide range of complex decision-making tasks that were previously out of reach for a machine to solve real-world problems with human-like intelligence. DeepMind published first successful algorithm about it [9].

In this work, we introduce a novel deep reinforcement learning method to predict the stock prices using knowledge graph and community aware sentiments. With the usage of knowledge graph, semantic convergence about stocks is inferred from comments. Thus, real-world relational objects (human) included to results and using social graphs to get more accurate opinions. It is known that influencers have a strong impact on investment ecosystem. Because of this reason, we gather comments from followers of people with a large number of followers (influencers) with the inclusion of social graph. In this way, we collect comments that contain the thoughts of people who are more relevant or knowledgeable about the related stock instead of collecting unrelated comments of random public users. In this way, we propose a more "live" methodology that includes real-world objects and relationships, understands sociological factors and concludes, accordingly. The proposed methodology presents deep reinforcement learning to forecast the stock price by employing knowledge graph-based sentiment signal. For this purpose, deep Q learning technique is used for deep reinforcement

learning methodology while Bidirectional Encoder Representations from Transformers (BerTurk) is utilized for sentiment analysis task. DBPedia [10] is also used to construct knowledge graph.

The rest of paper is presented as follows: In Section 2, studies in the literature related to financial applications of deep reinforcement methodologies are explained. Section 3 mentions on the proposed model and its details. Section 4 and 5 advert the experiment results and conclusion part, respectively.

2. Literature review

This section provides a brief summary of the state-of-the-art studies on deep reinforcement learning and its financial applications. In [11] Hu and Lin propose deep reinforcement learning model in order to eliminate essential research problems of policy optimization on finance portfolio management. They investigate the impact of recurrent neural network (RNN) models in order to observe the effects of former states and actions on policy optimization. After that, an available risk-oriented reward mechanism is constructed to appraise expected all rewards. Then, authors focus on integrating reinforcement learning and deep learning approaches in order to find an optimal policy. They report that each type reinforcement learning blended with deep learning method is capable to resolve policy optimization problem. In [12], Rundo introduces deep reinforcement learning approach in order to forecast financial trend in foreign exchange (FOREX) trading system using high frequency trading (HFT) algorithm. Thus, the author proposes the use of an algorithm based both upon long short-term memory network as a deep learning algorithm and on a reinforcement learning methodology for predicting the short-term trend in the currency FOREX market for the purpose of maximizing the return on investment in an HFT algorithm. Authors concludes the study that the introduced method is able to forecast the medium-short term trend of a currency cross based upon the trend of this historically with mean accuracy of nearly 85%.

In [13], Ye et al. present a reinforcement learning based portfolio management system by addressing two main challenges in portfolio management, namely data heterogeneity and environment uncertainty with their proposed model: State Augmented Reinforcement Learning (SARL) framework. The proposed SARL framework boosts the asset information with their price movement forecast as supplementary states in order to combine heterogeneous data and amplify durability against environment uncertainty. To prove the effectiveness of the SARS model, experiment are carried out on two real-world data sets, namely Bitcoin market data set and high-tech stock market with 7-year Reuters news articles. Experiment result demonstrate that proposed state augmentation approach with SARL provides new foresight and boosts considerably success over standard RL-based portfolio management methods and other traditional techniques. In [14], Xiao and Chen propose sentiment analysis-based reinforcement learning model to predict the stock return using only text data. Q-learning technique is performed as a reinforcement learning methodology in order to discover the optimal trading policy by learning the feedbacks from the market. Experiment results show that both of the machine learning method and the Q-learning technique excels the traditional model, logistic regression, without sentiment attributes. They conclude the paper that forecasting direction of stock price using only text data from Twitter sentiment is challenging but encouraging.

In [15], Li et al. propose a method based on a deep reinforcement learning methodology for trading stock and forecasting direction of stock price. Three different and conventional deep reinforcement learning techniques are utilized, namely deep Q-network (DQN), double deep Q-network (DDQN), dueling double deep Q-network (DDDQN). To show the effectiveness of deep reinforcement learning methodology, the historical daily prices and volumes of random ten stocks in all US-based stocks and exchange trade funds (ETFs) are chosen from NYSE, NASDAQ, NYSE MKT to construct the data set. Experiment results indicate that the usage of DQN for

direction of stock price outperforms other deep reinforcement learning techniques. In [16], Koratamaddi et al. present a new deep reinforcement learning methodology in order to construct an automated system for trading purpose. For this aim, five different models are employed namely, min variance analysis, mean variance analysis, deep deterministic policy gradients, adaptive deep deterministic policy gradients, adaptive sentiment-aware deep deterministic policy gradients. In addition to including historical stock price data, authors also investigate the impact of market sentiment comprising of the Dow Jones companies between 2015 February and 2018 February. Authors report that adaptive sentiment-aware deep deterministic policy gradients technique outperforms other models to orient investments. In [17], Chen and Gao introduce a deep reinforcement learning technique for automated stock trading. Deep Q-network (DQN) and deep recurrent Q-network (DRQN) are evaluated to automate stock trading system. In order to observe the efficacy of the proposed model, the S&P 500 ETF and its daily movements are employed as a data set. Authors state that DQN model performs better than DRQN technique to trade the stocks. In [18], Nan et al. introduce reinforcement learning model by blending with sentiments of news headline and knowledge graph for trading purpose. Deep Q-network is assessed as deep reinforcement methodology. To prove the effectiveness of the DQN network, stock data of Microsoft, Amazon, and Tesla and text data from the Reuters account on Twitter are gathered from January 2018 to December 2018. They conclude the paper that the reinforcement learning methodology outperforms other models in terms of profits.

3. Methodology

In this section, a brief summary of the methods, materials, and proposed framework are presented.

3.1. Bidirectional encoder representations from transformers (BERT)

Bidirectional Encoder Representations from Transformers (BERT) is a new generation word embedding model, which means bidirectional encoder representations. Unlike other word embedding models of the BERT model, it is designed to pretrained the data set in both layers in two directions and to condition the word in both right and left contexts [19]. The BERT model can be used to fine-tune with an additional output layer to create cutting-edge models without the important task of answering questions and language extraction. It is conceptually simple and empirically powerful [19]. In Figure 1, the architecture of BERT model is presented where the arrows indicate the information flow from one layer to the next. The T1, T2, T3 boxes at the top indicate the final contextualized representation of each input word. Input words are demonstrated with E that lies from 1 to n.

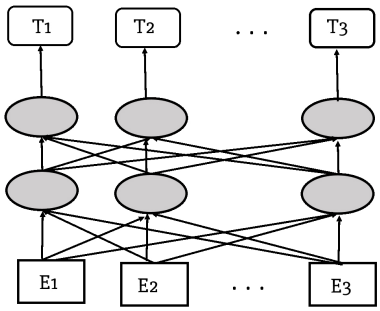

Figure 1. The architecture of BERT language model.

3.2. Deep Q-network (DQN)

The advantages of neural network and Q-learning algorithm are blended for DQN [20] approach. It is obtained by enhancing the function of experience replay from the transition of former state (experience) in the random sample training and accomplishing correlated and unsteady distribution data. In DQN, the present learned experience is demonstrated with the Q-value. The logic behind of DQN method is to learn the function of q-value and precisely forecast the q-value of each action in different states. The Q value is actually a score acquired by the agent owing to coaction with the environment and its self-experience, called as the target Q value. In summary: As a first step, samples are collected and stored in a replay buffer with present policy. Second, random sample batches of experiences are provided from the replay buffer, called as experience replay. Random sample experiences are employed in order to eliminate highly correlated experiences of past sequential experiences at this step. Thus, major bias issues are eliminated that can appear from correlated data. Third, the sampled experiences are employed in order to update the Q-network. For this purpose, the mean square error (MSE) between the target value of Q and our current output Q is minimized according to the Bellman equation in order to update the Q-network. Then, all steps (1–3) are repeated until the target q-value is reached.

3.3. Double deep Q-network (DDQN)

In deep Q and Q learning methodology, optimal Q value is employed in order to pick up and evaluate an action. For this reason, selecting an overestimated value can induce an overestimation of the real value of Q. Assessing the maximum overestimated values is implicitly evaluating the forecast of the maximum value. Hereby, this overforecast presents a maximization bias in learning methodology. This overforecast can be problematical because Q-learning contains bootstrapping, namely learning forecasts from forecasts. Thus, conventional DQN tends to notably overforecast action values by causing unsteady training procedure and low-quality policy. In order to avoid the overestimation problem of DQN, double deep Q-network approach is proposed by [21] by utilizing two individual Q-value forecaster, each of which is employed to update the another. With the utilization of these separate forecasters, Q-value forecasts have the ability to predict the actions picked up employing the opposite forecaster. Thus, the problem of maximization bias is prevented by delivering updates from biased forecasts.

3.4. Dueling double deep Q-network (DDDQN)

DDDQN [22] is constructed by aggregating double DQN and dueling DQN approaches. While DDQN is mentioned before the details about dueling DQN is given at first. In dueling DQN, there are two different estimates. The first one is to forecast for the value of a given state while second one is to predict for the advantage of each action in a state. Thus, q-value is denoted as how advantageous it is to be at that state and putting into an action at that state. Thence, q-value of an action at that state is dissociated as the total of the value of being at that state and the advance of putting into that action at that state. With decomposing procedure of the forecast, dueling DQN is capable to learn which states are worthy or not without having to learn the impact of each action at each state. In DDDQN, the mean advantage of all actions feasible of the state is subtracted from the output of aggregation layer. Thus, the problem of identifiability is eliminated that causes problem for back propagation that means not to find two elements. One forecasts the benefit for each action and the other forecasts the state value. This means that advantage function estimator is enforced have no advantage at the selected action.

3.5. Proposed framework

In this work, we propose to feed the reinforcement learning methodology with deep learning and sentiment analysis in order to predict the stock prices. In the study, the closing prices of three banking stocks in the BIST100 are estimated. The closing value of the stock on the previous day and the closing value of the stock on that day estimated by the proposed models, provide information about whether the direction of movement of the stock is upward, downward, or there is a change in the direction of the stock. When the closing value for that day estimated by the proposed models is subtracted from the closing value of the same stock on the previous day, if the difference is positive, the share moves up, and if it is negative, the share moves down. If the difference is zero, it means that there is no change in the direction of the stock.

With the increasing use of social media, individuals can be influenced by the ideas of influencers in these environments, and they can direct their investments accordingly by being inspired by the ideas of influencers. Traditional stock prediction is far from this reality and these real-world agents. The relations of the stock/company and opinions about the stock/company are of top priority for the proposed model. At least as important as, this is the relevance of the people who give their opinions to the investment. For this purpose, community finding algorithms are developed. Strong influencers can be determined in the investment community by using social graph analysis. These influencers are the strongest relational nodes which they connected other users with "interaction" and "following" relation types. In order to determine the impact of tweets on stock value, influence score is assigned to influencers. Thus, tweets are gathered from investment community with help of influencers and their interacted followers by considering highest influence score in order to perform sentiment analysis. Thus, investor community and influencer scores are constructed on the social graph.

The data scraped in the social graph consists of the usernames of the users in the social media and the follow-up relationship between each other. All these relationships are implemented as a graph database. The main reason for creating the social graph is to determine from which influencers we receive the tweets. Thanks to the relationships in the social graph, we define who our investor community consists of. Moreover, an information/knowledge graph is also employed on the expansion of the assets that are going to draw an opinion on. In the knowledge graph, the data and the relationship types between them were obtained from the structured Wikipedia, DBpedia. Thanks to knowledge graph structure taken from DBpedia, we can discover what kind of relations an entity has in the real world. In this way, while searching on tweets, we can feed on the knowledge graph and reach other entities that may affect it, apart from the name of company. Data resources of social and knowledge graphs are constructed directly from Twitter and DBpedia and there is no data manipulation in it. For this purpose, four basic types are employed such as locationCountry (country where the company is located), keyPerson (people important to the company), product (company's products), parentCompany (a parent company of the company), subsidiary (companies owned by the company). Tweets are collected through influencer listener services after gathering all keywords. In other words, we use them to search on tweets after determining keywords from the knowledge graph. These tweets include polarity extracted from the sentiment analysis model. Before performing sentiment analysis, preprocessing techniques are applied on tweets. These are cleaning the punctuation marks, the escape character removal, removing the website address and other links, hashtags, mentions, the face and other expression elements. The sentiment analysis model is developed utilizing the fine-tuned BERT model. Then, biased value is acquired by multiplying this polarity by the normalized coefficient value of sentiment analysis side which is given to reinforced learning as a signal. Then, sentiment score of intersected tweets is blended with stock values in order to feed environment

of the proposed model. In this way, a novel deep reinforcement learning based stock price prediction model is constructed using knowledge graph and community aware sentiments. After that, decision-making process is performed by employing DQN, proposed version of DQN (CA-DQN), DDQN, proposed version of DDQN (CA-DDQN), DDDQN, and proposed version of DDDQN (CA-DDDQN) to obtain buy, sell, and hold strategies. In Figure 2, the architecture of proposed framework is presented.

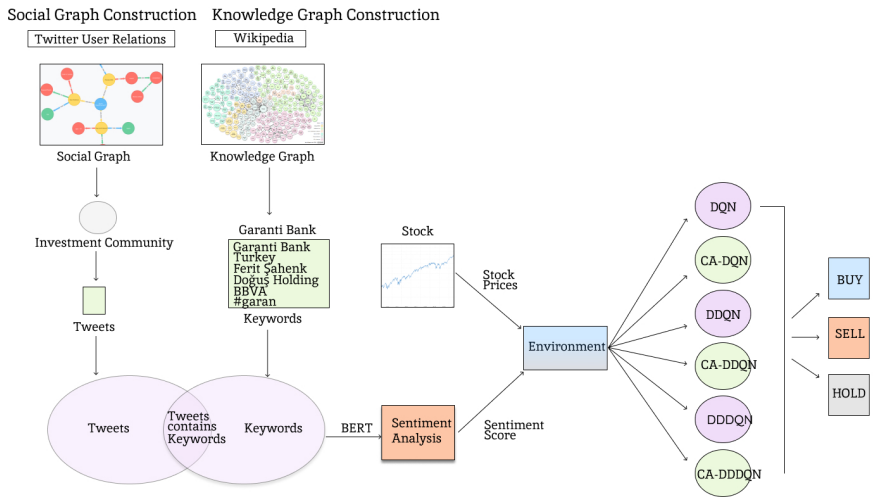

Figure 2. A general flowchart of the proposed deep reinforcement learning framework with the usage of knowledge graph and community aware sentiments.

3.5.1. Social graph and data set construction

In order to construct a social graph, Twitter data is gathered. To create a scattered distribution, the person who will start the scrape must be a high-profile account that appeals to the general audience. It is very important that the social graph obtained for determining various target groups has a scattered distribution. After the scrapping process is completed, a graph structure that expresses the general mass is obtained. There are many solutions for finding a community after constructing a social graph. Our solution will be to create a graph with the followers of the best influencers in the selected investment area and score it according to the number of followers in that graph. For this purpose, node score is calculated with follower count, named as influencer score. This approach is carried out on the community and extracted the influencer score of them. Because it is necessary to set a constraint to determine influencers with a large number of followers, having more than 100 followers within the community is the threshold for obtaining the top influencers in this study. As a result, 580 influencers are taken into consideration in this work. Scrape process of the tweets is performed between January 2015 and December 2020 utilizing Twint, which is an advanced Twitter scraping tool written in Python that allows for scraping Tweets from Twitter profiles without using Twitter's API. Similar to Twitter data set, the time series data is gathered between January 2015 and December 2020 using Yahoo Finance API.

3.5.2. Knowledge graph

Ontology is the database of terms connected with semantic relationships. They are often represented in a graph with entities and relationships. Knowledge graph, are more complex graphs where the entities are connected with features of their own. With these features, entities establish semantic connections between each other in a

way to define each other on the basis of features. In this work, we use DBPedia knowledge base that describes 4.58 million entities [10]. Linked real-word entities can establish relational affinity and find out which entity is relevant to which entity. In traditional methods, while this type of tweet analysis is associated with the keyword search, we will search for the keywords together with the relational keywords rather than just searching directly, and we will analyze the intersection of mentioned keywords and tweets that may be related. Considering one of data sets, Garanti Bank, in order to reveal close entities, we first determine the relationship types manually and add other entities to the keyword dictionary as a result of these types. The relationships consist of 5 basic types, namely location types, person types, parent companies, subsidiaries, and products. DBPedia relationships are presented for each type category as below:

- Location Types $\longrightarrow$ LocationCountry, RegionServed
- Person Types $\longrightarrow$ KeyPerson, KeyPeople
- Parent Companies $\longrightarrow$ parentCompany
- Subsidiaries $\longrightarrow$ subsidiary
- Products $\longrightarrow$ product

3.5.3. Sentiment analysis

Sentiment analysis is a method of determining whether a piece of writing, text, document is positive, negative or neutral. It is also known as idea mining, which examines the idea or attitude of a speaker. The common use of this technology is to discover how people feel about a particular subject. It refers to the use of intellectual mining, natural language processing, text analysis, computational linguistics and biometrics to systematically identify, measure and analyze emotional states and subjective information. Generally expressed emotion analysis aims to determine the attitude of a speaker, author or other subject regarding a topic, general contextual polarity or emotional response to a document, interaction or event. This attitude can be a trial or evaluation, or intended emotional communication. Our sentiment model is based on fine-tuned Turkish BERT cased model. Fine-tuned model is trained with e-commerce data set which size 50k. The evaluation accuracy value of the model, consisting of the test set of tweets, is resulted as 96%. The numerical value of the classes obtained as a result of the sentiment analysis is determined as 1 for positive, –1 for negative, and 0 for neutral class. Sentiment values are used in the scores of tweets sent that day. These tweet scores are also sent as a signal to the stock values of the same day.

3.5.4. Sentiment support score calculation

Before the sentiment analysis, we need to obtain the effect coefficient for each tweet. The effect coefficient is a type of coefficient associated with the keyword with each tweet, which influencer is tweeted, favorited count, mention count it is, retweet count. Interaction types can be different each other. Some types support values have more than others. "Retweet" is mentioned that by sharing the same thought, it kind of supports that idea. For calculating efficient score (ES), we use all type of interaction and influencer score (IS). Interaction bias (IB) is acquired by summing all biases from every interaction type. IB means the value showing the total impact value of the tweet. The sum of all interactions given allows to be obtained this value. However, the value strengthened with coefficient (retweet bias) is put directly instead of the number itself, so that how effective the retweet is reflected on IB. Retweet bias (RB) is a valuation made to strengthen the number of retweets and

demonstrate the number of retweets is much more effective than reply count, and like count. RB has stronger than other types, because of that we use biased retweet score instead of plain retweet score (RC). Unlike the efficient score, relational efficient score (ES_{rel}) is not derived directly from the bank's name, but from other entities from the bank's knowledge graph relationships. Since it is not directly associated with the bank, this value is reduced by a coefficient. Efficient score is directly derived from tweet source of bank name while ES_{rel} reflects the result of tweets obtained from entities in the knowledge graph. For this reason, ES calculation process can be different based on searched keyword type. In previous section, we talk about knowledge graph for relational entity to our main entity. If searched tweet's score, ES is divided by 4, its value will be reduced. Finally, for the main entity, Equation 4 is used for sentiment support with main entity.

$$RB = RCoe \times RC \tag{1}$$

$$IB = RB + LC + RepC \tag{2}$$

$$ES_{rel} = \frac{(IB + IS)}{RP} \tag{3}$$

$$ES = IB + IS, \tag{4}$$

where RCoe refers to retweet coefficient and RC is retweet count, LC denotes like count, RepC presents reply count, and RP is reduced parameter. After extracting efficient coefficient, each tweet is processed in the sentiment model. The positive, negative, and neutral values are multiplied by 1, –1, and 0, respectively. Thus, the coefficient values for sentiment analysis are obtained. However, values of coefficients are not at a level to combine raw data with stock data as numerical information. In cases where the number of daily tweets is high, it becomes very large compared to the stock value. In order to bring this value to a better level, the normalization is applied between 0 and 100 on a company-based basis for coefficient values. L2 normalization is carried out on the coefficient values in sentiment analysis side. The purpose of normalization is to merge the sentiment value with the time series data set, while the sentiment values have increased from the instantaneous data density to higher values. Therefore, these values were applied for both the train and the test set. The data that will come out of these data are again produced and normalized with sentiment values, and in this way the data becomes usable. In this way, it is determined how the sentiment is analyzed for that day and brought it to more meaningful data for stock values. On the other hand, no normalization or standardization methods have been applied on the time series data.

3.5.5. Markov decision process

In proposed deep reinforcement learning methodology, stock price prediction system is constructed by associating time series analysis with sentiment analysis results. Deep reinforcement learning can be summarized as creating an algorithm or an artificial intelligence (AI) agent that learns to interact directly with an environment by utilizing reward/penalty mechanism. In this way, the AI agent like a person learns the results of their actions (reward or penalty) rather than being taught explicitly. In the proposed system, the reward mechanism is a numerical value resulting from the sentiment analysis and the signals of some indicators, apart from the traditional time series analysis. It is observed that prediction of stock prices is a sequential decision-making process, as the investor would require to make investment options every day, one day after the other. Thus, the

issue of prediction of stock prices is modelled as a Markov decision process (MDP). Our current environment daily defines each state utilizing six variables:

1. Closing stock price on today's date.
2. Sentiment value towards the stock for today's date.
3. Average growth value of last 5 days.
4. Average sentiment value of last 5 days.
5. Average growth value of last 30 days.
6. Average sentiment value of last 30 days.

While (1) value is essential for maintaining the state of the agent (2), (4), and (6) ensure the sentiment information calculated as described in the previous section over a short time period (5 days) and a longer time period (30 days). In the reward formulation, there are three basic time intervals that are zero to five days, five to thirty days, and time interval greater than 30 days. The effect coefficient of daily values is set to 2 times higher than the last 5 days and 4 times higher than the last 30 days because the current issues have a greater effect on the stock market. Growth information is provided with the state of the agent (3), and (5) by calculating growth bias. It is effective in equal coefficients since there is no up-to-date growth rate on the stock. For (3), the average of the last 5 days is taken and subtracted from the closing value of that day. The difference is divided by the average of 5 days and growth is obtained according to the average. To find the effect of this rate on that day, a signal is obtained by multiplying by that day. After that, the reward is sent to the environment as a signal by blending time series data, sentiment analysis, and time series indicator in Markov decision process as mentioned before in order to get an action. In this way, a policy will be determined based on the time series analysis combined with both growth and sentiment bias.

In action space, the agent, the proposed stock prediction model, interacts with this environment on a per day basis. It has the option to take three actions:

- Buy a stock (BS)
- Sell a stock (SS)
- Do nothing/Hold (DNH)

In order to construct deep Q-Network (DQN) architecture, Q-network, and target network is employed with each with 3-hidden layers for the purpose of function approximation. Each hidden layer is constructed with 64 units and ReLU activation function. The input layer had six input nodes that correspond to the feature of each state. Finally, DQN architecture is completed with three nodes that correspond to the actions namely, BS, SS, and, DNH. In addition, the experience replay buffer size is set to 1000. In DQN networks, the experience replay is employed to learn from the previous experiences and is employed to store the previous states, actions, rewards, and next states. In this study, the data from the replay memory is sampled randomly and fed to the train network in small batch sizes for all DQN networks to eliminate overfitting problem. Moreover, early stopping is applied which is a procedure that aims to stop training at the level when performance on a test data set starts to degrade. Training procedure with DQN is performed by mini-batch gradient descent using Adam

optimizer [23]. The agent is interacted with the environment between January 2015 and January 2020. The agent in the form of a DQN is trained over 50 epochs. All versions of DQN networks are carried out the same experiment settings. The training procedure is carried out on Garanti Bankası, Türkiye İş Bankası, Akbank data sets. In Figure 3, the DQN network of proposed framework is presented.

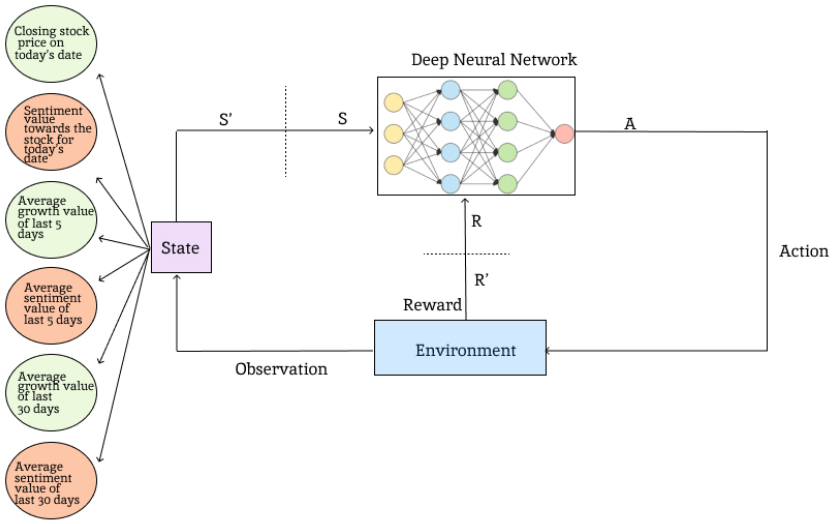

Figure 3. Markov decision process of proposed deep reinforcement learning framework.

4. Experiment results

The primary purpose of this work is that maintaining sentiment information to the agent on a daily basis would contribute to its performance and it would be able to ensure more profit by determining buy or sell signal. For this purpose, we compare deep Q-network with and without community analysis, double deep Q-network with and without community analysis, dueling double deep Q-network with and without community analysis approaches. In other words, it is proposed to compare the performances of an agent with sentiment data provided and another agent without any sentiment data provided. The following abbreviations are utilized for methods used in the experiments: DQN: deep Q-network, CA-DQN: deep Q-network with community analysis, DDQN: double deep Q-network, CA-DDQN: double deep Q-network with community analysis, DDDQN: dueling double deep Q-network, CA- DDDQN: dueling double deep Q-network with community analysis, GARAN: Garanti Bankası data set, ISCTR: Türkiye İş Bankası dataset, AKBNK: Akbank data set. All techniques are implemented on Google Colab environment provided free GPU utilization by Google. In this work, train-test profit scores in Turkish Lira, and Sharpe ratio are assessed as evaluation metrics. The train-test profit scores provide for the observation of the return on the stock while the Sharpe ratio is another evaluation metric that is frequently employed in trading as a means of assessing the risk adjusted return on investment. Profit calculation is the cumulative collection of the effects of the action applied at each step. Positions are updated in each purchase, and in each share sale, the value of the existing position is deducted for that day and is reflected as profit. It can be utilized as an evaluation metric to assess the performance of different techniques for the purpose of trading. It is computed as the expected return of a stock minus the risk-free rate of return (RF), divided by the standard deviation of the stock investment. RF value is calculated by taking the 12-month average deposits

opened on Turkish Lira with a term of up to 1 month in 2020 from the website of Central Bank of the Republic of Turkey. This value is assigned as 9.94. The Sharpe ratio is considered decent when it is 1. About 2 or higher is very good and 3 is evaluated as excellent. The best profit scores and Sharpe ratio results are represented in bold letters in Table 1. For Garanti Bank, the dictionary set keywords are "Turkey", "Ferit Şahenk (Founder)", "Doğuş Holding (Parent Company)", "BBVA (Parent Company)", "#garan (Exchange Code)", "Garanti Bank (Company Name)". For Akbank, the dictionary set keywords are composed of "Turkey", "Suzan Sabancı Dinçer (Founder)", "Personal Bank Services (Service Area)", "Investment Banking (Service Area)", "Mortgage (Service Area)", "#akbnk (Exchange Code)", "Akbank (Company Name)". For Isbank, the dictionary set keywords are constructed as "Turkey", "Investment Banking (Service Area)", "Special Banking (Service Area)", "#isctr (Exchange Code)", "İş Bankası (Company Name)". In Table 1, performance of different agents for Garanti Bankası, Türkiye İş Bankası, Akbank data sets are demonstrated in terms of train-test profit scores, and Sharpe ratios. It is clearly observed that the proposed framework by including community analysis generally exhibits higher performance compared to the traditional techniques such as DQN, DDQN, and DDDQN. CA-DQN outperforms both other versions of CA-based techniques and traditional reinforcement learning methodologies for AKBNK and ISCTR in terms of test profit scores while the best test profit score (186) is obtained with the utilization CA-DDDQN for GARAN data set.

Table 1. Performance of different agents for different stocks in terms of train-test profit scores and Sharpe ratios.

Model	**Evaluation metric**	**GARAN**	**AKBNK**	**ISCTR**
DQN	Train profit	7	-5	-17
	Test profit	78	73	40
	Sharpe ratio	1.73	1.64	0.75
CA-DQN	Train profit	9	138	184
	Test profit	133	**192**	**392**
	Sharpe ratio	2.32	**2.85**	**4.69**
DDQN	Train profit	1	3	4
	Test profit	15	49	31
	Sharpe ratio	0.45	1.12	0.56
CA-DDQN	Train profit	376	16	150
	Test profit	9	22	198
	Sharpe ratio	0.32	0.51	2.93
DDDQN	Train profit	23	10	3
	Test profit	10	10	20
	Sharpe ratio	0.36	0.18	0.42
CA-DDDQN	Train profit	1286	5	20
	Test profit	**186**	46	47
	Sharpe ratio	**2.67**	1.07	0.88

The test profit score order can be summarized for GARAN data set as CA-DDDQN > CA-DQN > DQN > DDQN > DDDQN > CA-DDQN. In summary, the best test profit score is obtained with CA-DDDQN for GARAN data set, CA-DQN for both AKBNK and ISCTR data sets. It is also observed that all stocks for all methods have positive return when test profit scores are considered.

The test profit of most stocks is higher than the train profit because the test set employs the optimal Q value of the training set in the training procedure. In unsupervised mode, the impact of employing the optimal

Table 2. Statistics of transactions for the best performed models on each data set.

Dataset	Buy	Hold	Sell	Transaction	Profit	Loss	Success
GARAN	281	38	24	343	22	2	91.66%
AKBNK	295	53	25	343	22	3	88.00%
ISCTR	232	65	46	343	40	6	86.95%

Q value is better than that in the process of exploring the optimal Q value. In general, even performance order of test profit score varies among data sets, the proposed CA-based proposed techniques perform generally well compared to the conventional models. Especially, DDQN with 0.45 and DDDQN with 0.36 Sharpe ratios exhibit a pretty poor policy as well, despite making profits for GARAN data set. Similarly, DDDQN with 0.18 for AKBNK data set and 0.42 for ISCTR data set demonstrates a terrible policy compared to the other policies in terms of Sharpe ratio.

In summary, CA-DDQN with 0.32 of Sharpe ratio for GARAN data set, DDDQN with 0.18 of Sharpe ratio for AKBNK and 0.42 of Sharpe ratio for ISCTR data sets learn a terrible policy and the Sharpe ratio of them is the least among all six approaches. It appears that the GARAN, AKBNK, and ISCTR stocks have a general ascending pattern even and because of this reason a poor policy can also generate profits with CA-DQN and DDDQN methods, despite making suboptimal strategies denoted by the Sharpe ratio. Moreover, not only did proposed models acquire the highest profits as stated earlier, but also their decision making is very well as confirmed by Sharpe ratios of 2.67 for GARAN, 2.85 for AKBNK, and 4.69 for ISCTR. As a result of Table 1, the results demonstrate that CA-based proposed techniques can effectively learn a profitable strategy from history data.

In Table 2, the statistics of the transactions are presented for the best performed model on each data set. The proposed model makes a total of 343 transactions by making one transaction every day for 1 year. Of these 343 transactions, 24 are sell transactions for GARAN data set. Twenty-two of the 24 transactions result in a profit and 2 of them result in a loss, and a total of 186 TL generates a profit at the end of 1 year. If any investor is encouraged to purchase 1 piece of GARAN stock at the closing price (11.03 TL) on the first day of January 2020 and keep the stock for 1 year and to sell it at the closing price (10.15) on the last trading day of December, investor incurs a loss of 0.88 TL (buy and hold strategy). However, the proposed model generates a profit of 186 TL for GARAN stock through its sales within 1 year. This is a proof that the model is successful. Moreover, proposed model does not trade with 1 share. Suppose, on the first day, it trades and makes a profit, and sells. If it makes a purchase transaction the next day, there will be a difference in the number of shares due to both the profit it has made and the price change that may occur in the stock. The change in the profit it receives is due to the addition of profit to the capital. Success metric is only possible to measure the successful operation of the model in sell transactions. After the sale, if a profit is made over the cost of the previously buy acts, this indicates that the model works positively, while in the case of loss, the model performs negatively.

Figures 4–6 show the reward and loss functions of the GARAN, AKBNK, and ISCTR stocks, respectively after the training by six models. Aforementioned before, reward specifies the target in the reinforcement learning problem. Reward function denominates to the overall reward caused by the modification of environmental state affected by the sequence of actions chosen at each time period. In other words, it is an instant reward that can evaluate the cons and pros of the actions. In addition, loss function is employed to forecast the grade of discrepancy between the actual value and the forecasted value of the model. The experiment compares the

loss functions of the six deep reinforcement learning models in order to evaluate and compare the economic advantages of these techniques. In Figures 4–6, (a), (b), (c), (d), (e), and (f) notations specify DQN, DDQN, DDDQN, CA-DQN, CA-DDQN, and CA-DDDQN models, respectively.

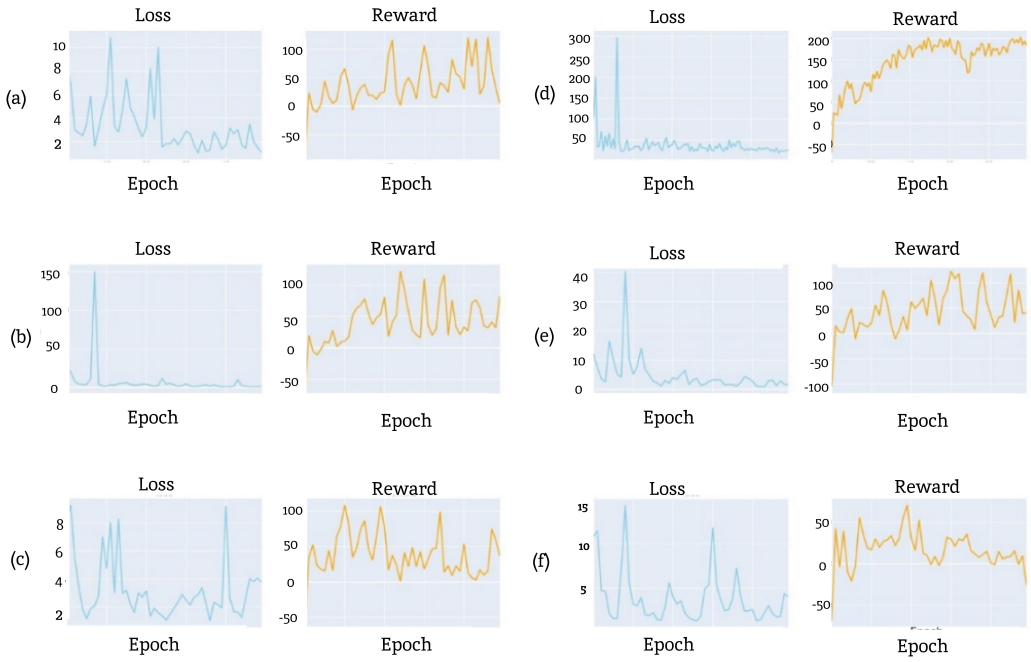

Figure 4. The loss-reward outputs of the proposed framework for the GARAN stock. Each letter represents a model as follows: (a) DQN, (b) DDQN, (c) DDDQN, (d) CA-DQN, (e) CA-DDQN, and (f) CA-DDDQN.

Figure 7 demonstrates an overview of the date-stock value of the best proposed models namely, CA-DDDQN for GARAN data set, and CA-DQN for AKBNK and ISCTR data sets in terms of buy, sell, and hold strategies. The grey color means that 'hold' operation is selected. The cyan line shows that the 'buy' is chosen. While the pink line demonstrates that the 'sell' is chosen at this time, 'hold' signifies the proposed model adopts 'hold-and-wait' behavior for the current position, i.e. neither buy nor sell. 'Buy' implies the trading model chosen by traders with a bullish behavior towards the potential stock trend. 'Sell' stands for the trading model selected by traders with a bearish behavior towards the potential stock trend.

When examined in Figure 7a, the proposed CA-DDDQN model gives generally 'sell' signal for the test data set at the beginning of 2020 when the value of the GARAN stock is at the top. The stock declines in its subsequent movements and the proposed model produces 'buy' signal in the declines. In this case, the investor will not have missed the opportunity to buy the stock in declines by buying the stock piecemeal where there is a buy signal. In Figure 7b, the proposed CA-DQN model generally gives 'sell' signal where AKBNK stock makes a double top movement for the test data set in the mid and late of 2020. Therefore, in the case of falls that occur after seeing the double top of the stock, the loss is not observed in the budget of investor, mostly. Similarly, where the stock makes a double top, the proposed model generates a 'buy' signal from the bottom value of the stock. This, in turn, allows the investor to cost at the lowest value of the stock. In Figure 7c, the proposed CA-DQN model generally gives 'sell' signal for the test data set at the beginnings of 2020 and 2021 when the value of the ISCTR stock is at the top. Furthermore, at the mid of 2020 and the beginning of 2021,

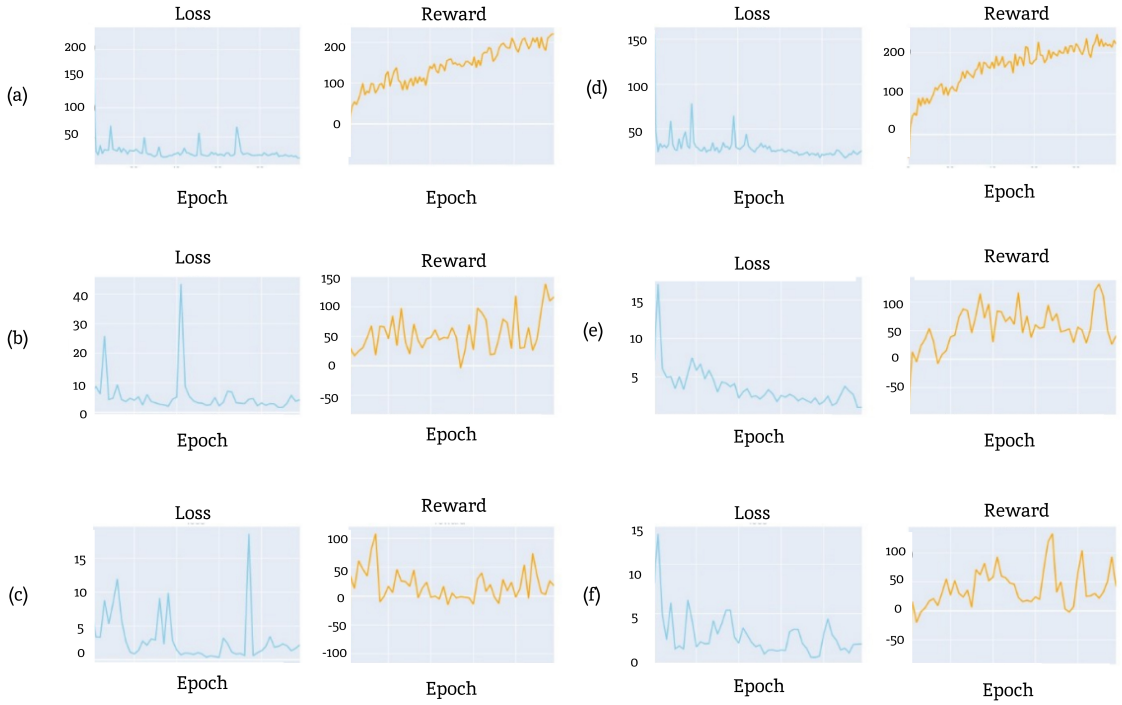

Figure 5. The loss-reward outputs of the proposed framework for the AKBNK stock. Each letter represents a model as follows: (a) DQN, (b) DDQN, (c) DDDQN, (d) CA-DQN, (e) CA-DDQN, and (f) CA-DDDQN.

when the ISCTR stock makes a double top, the proposed model produces a 'buy' signal from the bottom value of the stock similar to AKBNK stock movement. Similarly, it produces buy signal in early 2020, treating each decline as a buying opportunity.

In Figure 8, performance of the best agent for GARAN, AKBNK, and ISCTR stocks are presented. Figure 8 shows the close price forecasting line chart of the above three stocks, where the black line indicates the original close price, and the red line demonstrates the predicted close price. In every subfigure, x-coordinate represents the date, while the y-coordinate represents the corresponding closing price of each stock. For all data sets, we evaluate test data set, which spans from January 1, 2020 to December 31, 2020 to demonstrate the contribution of CA-based novel deep reinforcement learning approaches for predicting the price of stocks in Borsa İstanbul.

CA-DDDQN performs very well in GARAN data set while CA-DQN approach is almost overlapped to the closing price, which means CA-based techniques are efficiently capable to learn a profitable strategy from history data. This means that including the knowledge graph and community aware sentiments remarkably contributes the prediction of stocks prices to orient the investments for investors, analysts, researchers. Furthermore, blending knowledge graph and community aware sentiments with deep reinforcement learning methodology becomes the stock price estimations more stable and robust compared to the traditional reinforcement learning approaches. Considering the prediction performance of and flexibility of the proposed framework, it can be also applicable other investment tools such as digital currencies, stocks, mineral commodities such as gold, silver, bonds, funds and such products. That is why the model we propose for investors, researchers and analysts becomes even more attractive.

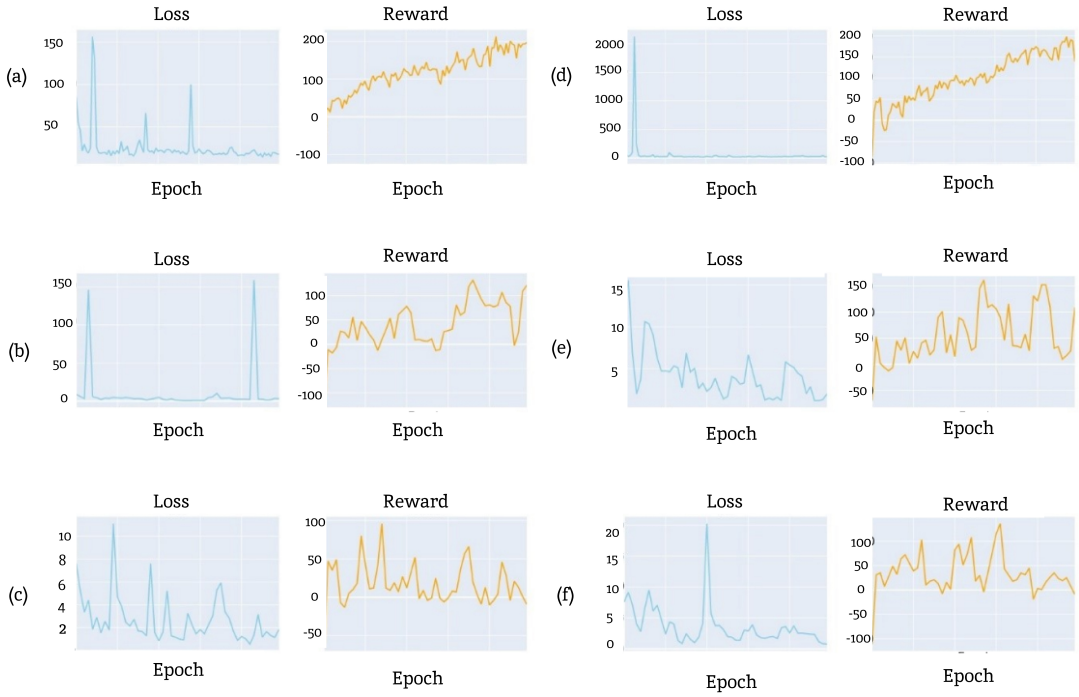

Figure 6. The loss-reward outputs of the proposed framework for the ISCTR stock. Each letter represents a model as follows: (a) DQN, (b) DDQN, (c) DDDQN, (d) CA-DQN, (e) CA-DDQN, and (f) CA-DDDQN.

Figure 7. Buy, sell, and hold simulations of best proposed models for each data set (a) CA-DDDQN for GARAN data set, (b) CA-DQN for AKBNK data set, and (c) CA-DQN for ISCTR data set.

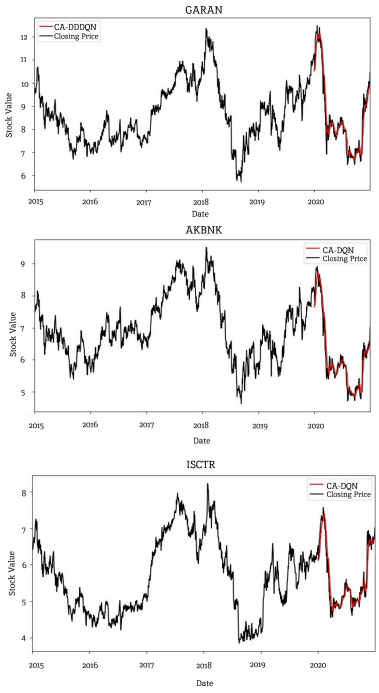

Figure 8. Comparison of the best proposed approaches and closing price on the GARAN, AKBNK, and ISCTR stocks.

5. Discussion and conclusion

In this work, we propose a novel technique that is based on deep reinforcement learning methodologies for the prediction of stock prices blending sentiments of community and knowledge graph. For this purpose, a social knowledge graph of users is firstly constructed thereby evaluating relations between connections. Then, time series analysis of related stocks and sentiment analysis is combined with deep reinforcement methodology. Turkish version of Bidirectional Encoder Representations from Transformers (BerTurk) is utilized to analyze the sentiments of the users while deep Q-learning methodology is used for the deep reinforcement learning side of the proposed model to construct the deep Q-network. Deep Q-network (DQN), proposed deep Q-network with community analysis (CA-DQN), double deep Q-network (DDQN), proposed double deep Q-network with community analysis (CA-DDQN), dueling double deep Q-network (DDDQN), proposed dueling double deep Q-network with community analysis (CA-DDDQN) approaches are employed in the experiments to compare the prediction performance of each model. In order to demonstrate the effectiveness of the proposed model, Garanti Bank (GARAN), Akbank (AKBNK), Türkiye İş Bankası (ISCTR) stocks in Borsa İstanbul are used as data sets.

Experiment results show that the proposed novel models generally demonstrate higher performance compared to the conventional techniques such as DQN, DDQN, and DDDQN. CA-DQN outperforms both other versions of CA-based techniques and traditional reinforcement learning methodologies for AKBNK (192) and ISCTR (392) in terms of test profit scores while the best test profit score (186) is obtained with the utilization CA-DDDQN for GARAN data set. The test profit score order can be summarized for GARAN data set as CA-DDDQN$>$ CA-DQN$>$ DQN$>$ DDQN$>$ DDDQN$>$ CA-DDQN. The test profit score order can be summarized for AKBNK data set as CA-DQN$>$ DQN$>$ DDQN$>$ CA-DDDQN$>$ CA-DDQN$>$ DDDQN. The test profit score order can be summarized for ISCTR data set as CA-DQN$>$ CA-DDQN$>$ CA-DDDQN$>$

DQN> DDQN> DDDQN. In summary, the best test profit score is obtained with CA-DDDQN as 186 for GARAN data set, CA-DQN for both AKBNK and ISCTR data sets, 192, and 392, respectively. It is also seen that all stocks for all methods have positive return when test profit scores are considered. The test profit of most stocks is higher than the train profit because the test set utilizes the optimal Q value of the training set in the training procedure. Furthermore, CA-DDQN with 0.32 of Sharpe ratio for GARAN data set, DDDQN with 0.18 of Sharpe ratio for AKBNK data set and 0.42 of Sharpe ratio for ISCTR data set learn a poor policy and the Sharpe ratio of them is the least when compared with other techniques. It seems that the GARAN, AKBNK, and ISCTR stocks have a general rising trend even and owing to poor policy can also generate profits with CA-DQN and DDDQN techniques, spite performing suboptimal strategies identified by the Sharpe ratio. Additionally, not only did proposed models achieve the highest profits aforementioned above, but also their process of decision making is very well as demonstrated by Sharpe ratios of 2.67 for GARAN, 2.85 for AKBNK, and 4.69 for ISCTR. As a result, experiment results indicate that CA-based proposed models can efficiently learn a profitable strategy from history data.

In [16], Koratamaddi et al. introduce a new deep reinforcement learning methodology to build an automated system for trading purpose. Min variance analysis, mean variance analysis, deep deterministic policy gradients, adaptive deep deterministic policy gradients, adaptive sentiment-aware deep deterministic policy gradients techniques are utilized. Authors report that adaptive sentiment-aware deep deterministic policy gradients model surpasses others with 2.07 of Sharpe ratio. Similarly, proposed knowledge graph and community aware sentiments with deep reinforcement learning methodology exhibits superior performance compared to the traditional deep Q-networks, in this study. CA-DDDQN with 2.67 Sharpe ratio for GARAN, CA-DQN with 2.85 Sharpe ratio for AKBNK, and CA-DQN with 4.69 Sharpe ratio for ISCTR are obtained in our experiments. In [18], Nan et al. present reinforcement learning model by combining with sentiments of news headline and knowledge graph for trading purpose. Deep Q-network is evaluated as deep reinforcement methodology. To demonstrate the efficiency of the DQN network, stock data of Microsoft, Amazon, and Tesla and text data from the Reuters account on Twitter are collected. They inform that the proposed model with sentiment data accomplishes 2.432 for Microsoft, 2.212 for Amazon, 1.874 for Tesla in terms of Sharpe ratio evaluation metric while the agent without sentiment achieves -1.357 for Microsoft, 1.487 for Amazon, 0.926 for Tesla in terms of Sharpe ratio. In our study, we also use three stocks namely, GARAN, AKBNK, and ISCTR. Proposed knowledge graph and community aware sentiments with deep reinforcement learning methodology in our work exhibits 2.67 for GARAN, 2.85 for AKBNK, 4.69 for ISCTR in terms of Sharpe ratio. Experiment results demonstrate that the inclusion of knowledge graph and community aware sentiments into deep Q-network and its versions contributes the performance of traditional DQN networks as in the literature studies. As a future work, we also intend to blend transfer learning strategy for predicting stock prices by extending investment tools.

References

[1] Leung MT, Daouk H, Chen AS. Forecasting stock indices: a comparison of classification and level estimation models. International Journal of forecasting 2000; 16 (2): 173-190.

[2] Kumar M, Thenmozhi M. Forecasting stock index movement: A comparison of support vector machines and random forest. In: 9th Capital Markets Conference; India, 2006. pp. 1-16.

[3] Abu-Mostafa YS, Atiya AF. Introduction to financial forecasting. Applied intelligence 1996; 6 (3): 205-213.

[4] Tan TZ, Quek C, Ng GS. Biological brain-inspired genetic complementary learning for stock market and bank failure prediction. Computational Intelligence 2007; 23 (2): 236-261.

[5] Goonatilake R, Herath S. The volatility of the stock market and news. International Research Journal of Finance and Economics 2007; 3 (11): 53-65.

[6] Gabielkov M, Legout A. The complete picture of the Twitter social graph. In: 2012 ACM Conference on CoNEXT Workshop; Nice, France 2012: 19-20.

[7] Sutton RS, Barto AG. Reinforcement learning: An introduction. 2nd ed. Cambridge, MA: MIT press, 2018.

[8] Holcomb SD, Porter WK, Ault SV, Mao G, Wang J. Overview on deepmind and its alphago zero ai. In: 2018 International Conference on Big Data and Education; Hawaii, USA 2018:67-71.

[9] Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J et al. Human-level control through deep reinforcement learning. Nature 2015; 518 (7540): 529-533.

[10] Auer S, Bizer C, Kobilarov G, Lehmann J, Cyganiak R et al. DBpedia: A Nucleus for a web of open data. In: Asian Semantic Web Conference; Busan, Korea 2007: 722-735.

[11] Hu YJ, Lin SJ. Deep reinforcement learning for optimizing finance portfolio management. In: 2019 IEEE Amity International Conference on Artificial Intelligence; Dubai, UAE 2019: 14-20.

[12] Rundo F. Deep LSTM with reinforcement learning layer for financial trend prediction in FX high frequency trading systems. Applied Sciences 2019; 9 (20): 4460-4478.

[13] Ye Y, Pei H, Wang B, Chen PY, Zhu Y et al. Reinforcement-learning based portfolio management with augmented asset movement prediction states. In: 2020 AAAI Conference on Artificial Intelligence; New York, USA 2020: 1112-1119.

[14] Xiao C, Chen W. Trading the Twitter Sentiment with Reinforcement Learning. arXiv preprint arXiv:1801.02243, 2018.

[15] Li Y, Ni P, Chang V. Application of deep reinforcement learning in stock trading strategies and stock forecasting. Computing 2019; 102: 1305–1322.

[16] Koratamaddi P, Wadhwani K, Gupta M, Sanjeevi SG. Market sentiment-aware deep reinforcement learning approach for stock portfolio allocation. International Journal of Engineering Science and Technology 2021; 24 (4): 848-859.

[17] Chen L, Gao Q. Application of deep reinforcement learning on automated stock trading. In: 2019 IEEE 10th International Conference on Software Engineering and Service Science; Beijing, China 2019: 29-33.

[18] Nan A, Perumal A, Zaiane OR. Sentiment and knowledge based algorithmic trading with deep reinforcement learning. arXiv preprint arXiv:2001.09403, 2020.

[19] Tenney I, Das D, Pavlick E. BERT rediscovers the classical NLP pipeline. arXiv preprint arXiv:1905.05950, 2019.

[20] Mnih V, Kavukcuoglu K, Silver D, Graves A, Antonoglou I et al. Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602, 2013.

[21] Van Hasselt V, Guez A, Silver D. Deep reinforcement learning with double q-learning. In: 2016 AAAI Conference on Artificial Intelligence; Arizona, USA 2016: 2094-2100.

[22] Wang Z, Schaul T, Hessel M, Hasselt H, Lanctot M et al. Dueling network architectures for deep reinforcement learning. In: In International Conference on Machine Learning; New York, USA 2016: 1995-2003.

[23] Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.