



TFCMA-DRL: Tolerant Flexible Coordinated Multi-Agent Deep Reinforcement Learning for Prediction of Future Stock Price Trends from Multi-Source Data

Chinnasamy Bhuvaneshwari^{1*} Raman Beena¹

¹*Department of Computer Science, Kongunadu Arts and Science College, Coimbatore, India*

* Corresponding author's Email: bhuvaneshwari.c2020@gmail.com

Abstract: Prediction of future stock price volatility influence the economic empowerment in major businesses. Machine learning and deep learning algorithms were the most efficient techniques used for prediction by extracting public opinions and events. However, the extraction of sentiments and events are less cohesive because of sparse learning. To tackle these issues, the Tolerant Flexible Coordinated Multi-Agent Deep Reinforcement Learning (TFCMA-DRL) model is proposed in this paper. This model utilizes Extreme Learning Machine (ELM) for extracting the tweet sentiments and Restricted Boltzmann Machine (RBM) for the event extraction from news articles. Finally, TFCMA-DRL learns the stock price patterns through these indicators and forecasts the future price trends. Evaluations on real-time data collected from Twitter, News website and Stock exchange for 12 months period. The results highlight the efficiency of the TFCMA-DRL model which achieved promising results of 96.67% accuracy, 99.5% precision, 96.67% recall, 98.305% F1-score values and also minimized time complexity.

Keywords: Stock market forecasting, Stock prediction, Sentiment analysis, Event extraction, Extreme learning machine, Restricted boltzmann machine, Tolerant flexible coordinated multi-agent deep reinforcement learning.

1. Introduction

Stock market is the accumulation of investors who purchase or sell the stocks representing the partial proprietorship rights on business entities listed in the public stock exchanges [1]. The equity crowd funding platforms and public stock exchanges allow such investments. The investment strategy is always depended on choosing the rising priced stocks to be purchased while selling those falling priced stocks. This strategy included various permutations and combinations of factors that provide exact prediction of these stock price fluctuations [2]. However, the prediction process is highly time consuming and complex for manual operations. Stock market prediction methods have been designed for ensuring the automatic and fast analysis of the factors and predicting the future stock price trends. Determining the future value of a stock depends on many stock price indicators or features that include public opinions, government policies, global stock index performance and many other internal and external factors [3]. Efficient

market hypothesis (EMH) has been predominantly utilized for revealing the factors and thus analysing the future trends [4]. Although the financial academicians prefer EMH, they also suffer from large deviations than actual stock values. With the advancements in computing field, the text mining algorithms with statistical and machine learning algorithms have been utilized largely in stock prediction. Artificial neural networks (ANN) [5], support vector machine (SVM) [6], and Random forests [7] are some of the most popular methods used for stock value prediction application. Most of these methods performed effectively for stock prediction but the performance was limited as their learning of stock price patterns still suffer from sparse representation.

Traditional stock price prediction models were based on the single-source data which means they depend only on a single set of features, often limited by solitary sentiments. This type of learning process provides high accuracy but is only one-dimensional in terms of the indicator learning [8]. This greatly reduces the effectiveness of the prediction model and limits its usage for real stock value prediction.

To alleviate these limitations, the multi-source data based prediction models were developed [9]. Although they provided efficient performance for real stock price prediction, the only limitation was the complexity it incurs for learning the multi-source indicators. Deep learning models have been employed in many applications for their extensive learning with less time complexity [10]. Hence they were preferred for stock price variation analysis and improved the accuracy through the deep learning of the stock related data. Still, most studies employed only one source data along with the stock quantitative data. Some authors used news and tweets data as multi-source data and achieved high prediction performance. But through extensive analysis, it has been found that the cohesiveness between the news events and the tweet sentiments is very limited. Extracting the story like cohesion features can help in improving the prediction. These suggestions are considered and an efficient stock price prediction model is developed.

In this paper, Coordination policy is integrated with Multi-Agent Deep Reinforcement Learning to form TFCMA-DRL based stock price forecasting model for multi-source data. ELM and RBM are used for extracting the maximum relation mapping based tweet sentiments and news events, respectively. Although, previous studies have extracted stock related events and public opinions from tweets for stock price forecasting, the cohesion between the indicators is less. To enhance the cohesion between the indicators, the proposed TFCMA-DRL model employs the coordinated learning for the sentiments and events extraction. The rest of article is organized as: Related works in Section 2. Proposed TFCMA-DRL based stock price forecasting model in section 3. The results are presented in section 4 followed by the conclusion of this paper in section 5.

2. Related works

Recently, many studies utilized the multi-factor and multi-source stock prediction models. The use of deep learning algorithms for stock price prediction has also seen massive increase. Yang et al. [11] presented multi-source analysis based stock forecasting model using multifactor and candlestick models. By selecting moving average (MA), relative strength index (RSI), and other technical indicators, this model enhances the prediction reliability. However, the noise factors are conveniently neglected in this model. Idress et al. [12] employed Auto Regressive Integrated Moving Average (ARIMA) model to predict the stock market changes

from the time series data. This model provides the prediction free from autocorrelation with accepted deviation of 5% mean percentage error which is efficient for many real world applications. However, ARIMA is a univariate model that cannot exploit the explanatory variables. Nayak et al. [13] developed artificial chemical reaction optimization based functional link network (ACFLN) for stock prediction with less complexity and faster convergence. Although this model has high prediction accuracy for short term, medium term and long term stocks with less uncertainty, it has limited performance in handling multi-source indicators. Tang et al. [14] presented a deep hierarchical strategy of Wasserstein generative adversarial network (WGAN) for pre-judging trend regression and deep deterministic policy gradient (DDPG) for forecasting. Although this model achieves high robustness, it has limitations in optimizing the WGAN for prejudging regression coefficients. Chandar [15] proposed a fusion model of wavelet-adaptive network-based fuzzy inference system (WANFIS) in which discrete wavelet transforms (DWT) decomposes the time series data for stock forecasting. However, this model has fixed wavelet coefficients that might degrade the adaptiveness and scalability. Xu et al. [16] introduced Unrestricted Mixed Data Sampling Support Vector Regression (UMIDAS-SVR) by feeding the mixed frequency market sentiments into SVR to tackle the nonlinear relationship and mixed frequency data problems. This method provided highly accurate prediction with better mean absolute error (MAE), root mean square error (RMSE) and effectively handles the non-linearity issue. Irrespective of these benefits, UMIDAS-SVR has considered only market sentiments while neglecting other multi-source sentiments.

Hoseinzade and Haratizadeh [17] introduced stock forecasting model for multi-source data using convolutional neural networks (CNN) and achieved high accuracy with the automatic extraction of high level features. However, the CNN prediction model has extensive structure designing process which is a huge challenge for smaller stock markets. Chung and Shin [18] developed Genetic algorithm-optimized convolutional neural network (GA-CNN) for high accurate stock prediction. However, GA increases the computation power in multiple iterations. Wen et al. [19] introduced a forecasting approach using CNN with sequence reconstruction by leveraging motifs to simplify the noisy stock time series data. This approach provided macroscopic pattern identification to improve the forecasting accuracy but it also increased the computation time.

Li et al. [20] presented stock market forecasting model using two-layer LSTM network which sequentially learns the historical stock price and news sentiments to improve the prediction reliability. However, this model extracts full-text sentiments from the news which reduces the learning of impacts in stocks rather than events.

Zhang et al. [21] developed multi-instance learning (MIL) model for extracting the knowledge from social media and news articles for forecasting the stock prices. Although it provided high accurate results, they do not perform efficiently when precise classes are not defined. Previously, TMA-DRL [22] and TFMA-DRL [23] models were developed to tackle these issues through efficient deep learning and advanced sentiment and event extraction models. Irrespective of their efficiency, the forecasting approaches in literature have also highlighted their shortcomings. One important limitation is the appropriate model for completely learning the multi-source data. The other vital drawback is the limited inference of the stock price patterns and different indicators. Hence considering these limitations, the proposed approach TFCMA-DRL has been designed to accurately forecast the future trends in the stock prices.

3. Methodology

The proposed stock market forecasting model consists of three main phases: Data collection, Extraction of sentiments and events, and learning the stock price patterns for prediction as shown in Fig. 1. Table 1 presents the list of notations used in this paper and their definitions.

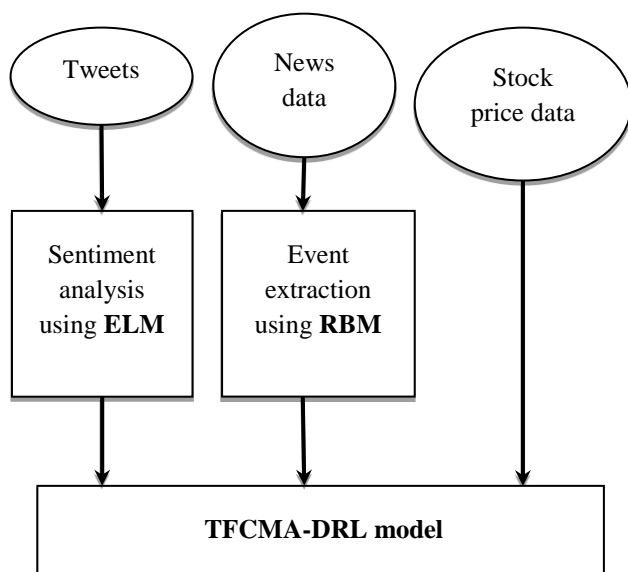


Figure. 1 TFCMA-DRL based stock forecasting model

Table 1. Notations and definitions

Notation	Definition
H	Hidden layer output matrix
$g(\cdot)$	activation function
b, b_i	Bias of ELM
w, w_i	Input weights
x, x_j	Feature weights
β	Weight vector
O	Output of ELM
H^\dagger	Moore–Penrose generalized inverse of H
l	Class labels
H_{Rest}	Testing data H matrix of ELM
O_{Rest}	Testing data output of ELM
a_t	action
θ	weights constraint of Q-networks
r	Outcome adjustment factor
γ	output weight of hidden layers
s_t	state of restructured Q-function
θ^-	target network weights
Q	output of deep learning network
$(\phi(s), a)$	heat value Temp and hash-keys
t	time
(s, a)	state-action pair
a^*	joint action with communication
a_i	action taken by individual agents
a_g	communicated actions between the agents
x_{it}	agents own observation
f	Classifier set
$G(z)$	global reward of unified state z
z_i	system state without agent i
u_i	counterfactual term
N	Number of agents
R	global reward signal
$Q_i(s_{it}, a_{it})$	state-action pair for agent i at time t
α	Learning rate

3.1. Data collection

The proposed TFCMA-DRL model has been employed using multi-source data from three different sources. The three source data are tweets, news articles and stock index prices collected at the same duration from October 1, 2018 to September 30, 2019. This 12 month data is split into training (first 11 months) and testing data (last month data).

Stock price data: Historical stock price data of BSE 500 and NSE 500 were gathered at BSE and NSE stock indexes for the 1year duration. It includes the beginning and ending price, highest and lowest stock values, revenue and trade bonuses of top 500 listed companies.

News data: The stock market related news including the government announcements was extracted from <https://www.moneycontrol.com>. It includes the date and time of news publication, headlines and news preamble from 47,652 news articles during the specified period. The data with non-English sentences are eliminated and the evaluation dataset was formed with 10,026 news articles.

Tweets: 176248 Public tweets were collected using 50 stock market related keywords. After initial screening, the non-English tweets are eliminated and the evaluation dataset was created with 66,542 tweets.

The collected data are pre-processed to minimize the missing values and noise problems. The stock price data are historical data and hence the noise must be removed significantly to reduce the negative impacts. The news and tweets data are applied with the NLP tools for language modelling. Apart from these processes, the tweets are processed additionally by removing the other language tweets and only the English tweets are retained.

3.2. Event extraction from news headlines

Hierarchical news is the highly interlinked news events that contain almost every correlated factor that impact a certain event. The cohesiveness between the news articles will impact the interrelated events. The restricted Boltzmann machine (RBM) [24] is used for the extraction of event features. First, a pre-trained vector will be the output of RBM while sentence2vec takes it as input and returns the event feature representations.

The syntactic structures of the news headlines are obtained through the text parser named HanLP. The core words are mapped into the hidden layer of RBM that extracts the events accurately. The standard RBM extracts the events through the hidden variables relatedness and the spatial relationships. The two layers of the RBM are modelled with the input and output of hidden layer. The input layer is mapped with the encoded m-dimensional vector and is used to estimate the n-dimensional vector of the hidden layer. The tree centred sentence constructions outlining the object and subjects and the essential words are associated to event representation. The visible units are independent for each other. This allows exploring the new structure of RBM with distinct feature models. These kinds of hidden feature models contribute the covariance structure of RBM and improve the probabilistic pooling and penalty term objective functions. The obtained structures are

correlated in such a way that the occurrence of an event possibly will trigger another related event with high probability. Employing this model enhances the maximum cohesiveness between the news articles. Based on this model, the output obtained from the RBM is given to the sentence2vec model for training the TFCMA-DRL.

3.3. Sentiment extraction from tweets

The major challenge in handling the tweets into multi-class classification is the class imbalance data. Extreme Learning machine (ELM) [25] and its variants are highly effective in handling this problem. To overcome the sparsity problems for tweet sentiment extraction, the ELM is employed in this study. The ELM algorithm is offered as an improved multilayer deep illustration network that increases the feature representation ability. The ELM initializes the neural network weights and biases arbitrarily. Then the weights of hidden layer are computed mathematically. The matrix for the output of the hidden layer H can be modelled as

$$H = \sum_{i=1}^N g(w_i \cdot x_j + b_i) \quad (1)$$

Here g denotes the activation function, b_i denotes the bias and w_i and x_j are the input weight and feature weight values, respectively. The output equation of the ELM can be simplified as

$$H \beta = O \quad (2)$$

Where β denotes the weight vector and O denotes the output of the ELM with the sigmoid function. This means the value of β is calculated as

$$\beta = H^\dagger O \quad (3)$$

Where H^\dagger denote the Moore–Penrose generalized inverse of H.

For applying the ELM for sentiment classification, the features of tweets are extracted and the weight values are computed. For example, a sample problem is provided in this section and the computed weights of tweets in training phase are given as

$$x = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \text{ and } l = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix} \quad (4)$$

Here l denotes the class labels and x indicate the input feature weights for the training tweets with 4 tweets (rows) and 6 features (columns) with each row having a document vector for tweets. The next step includes the random initialization of input weights (w) and biases (b).

$$w = \begin{bmatrix} 0.78 & 0.41 & 0.37 & 0.74 & 0.37 & 0.21 \\ 0.85 & 0.46 & 0.76 & 0.36 & 0.44 & 0.13 \\ 0.73 & 0.66 & 0.17 & 0.95 & 0.72 & 0.26 \\ 0.55 & 0.43 & 0.05 & 0.11 & 0.71 & 0.14 \end{bmatrix} \quad (5)$$

$$b = \begin{bmatrix} 0.98 \\ 0.4 \\ 0.21 \\ 0.3 \end{bmatrix} \quad (6)$$

The values provided in Eq. (5) and (6) are manually generated to illustrate the sentiment extraction process by ELM. In implementation, these values are generated automatically through the random initialization process. The initial values of (w) and (b) are significantly important in tuning the optimal structure of ELM. Hence they must be selected initially such that they do not exceed 1.

Then the value of H and β can be computed using Eq. (1) and (2) as follows:

$$H = \begin{bmatrix} 0.89 & 0.87 & 0.85 & 0.85 \\ 0.82 & 0.80 & 0.78 & 0.77 \\ 0.79 & 0.77 & 0.83 & 0.72 \\ 0.61 & 0.73 & 0.81 & 0.70 \end{bmatrix} \quad (7)$$

$$\beta = \begin{bmatrix} -57.693 & 482.6 & -116.98 & -299.08 \\ 66.38 & -569.61 & 118.394 & 371.66 \\ 5.5388 & 64.786 & 0.4075 & -68.69 \\ -10.575 & -23.298 & 9.165 & 23.614 \end{bmatrix} \quad (8)$$

After estimating these values, the training stage is finished and then the testing phase begins. In testing, each tweet has separate vector and feature weights $x = [1 \ 0 \ 0 \ 0 \ 0 \ 0]$ with class label $l = [3]$. The H matrix can be estimated by applying the training steps which is given as

$$H_{Test} = \begin{bmatrix} 0.85 \\ 0.77 \\ 0.72 \\ 0.70 \end{bmatrix} \quad (9)$$

The output of ELM can be computed as $O_{Test} = H_{Test}\beta$ given by

$$O_{Test} = \begin{bmatrix} -1 \\ -0.99 \\ -1 \\ 1 \end{bmatrix} \quad (10)$$

Here the values in O_{Test} are analysed and found that the last value is the largest and hence the tweet belongs to that class ($l = [3]$). Similarly, the sentiments of all tweets can be determined and they are utilized as the stock price indicators.

3.4. TFCMA-DRL

The proposed TFCMA-DRL model incorporates the coordination policies to the tolerant and flexible multi-agent system to learn the multi-source sentiments and events with high interrelation. The tolerance policy is applied to the flexible multiple agents to evade the relative overgeneralization. Similarly, the coordination policy [26] is added to increase co-ordination between agents by defining a communication mechanism. The coordinated model learns the slowest but is stable and converges to near perfect performance. These properties can greatly enhance the learning process of combined opinions, events and the quantitative data for the analysis of the stock market data.

Let θ be the weights constraint of Q-networks, a_t is the action, r is the adjustment factor value, γ is the output weight of the hidden layers and s_t is the state of the restructured Q-function. The network has three convolutional layers and two fully connected layers. For stability, the target network and flexible network are included; whose weights are updated after each round to match the objectives. The objective function is modelled as

$$L(\theta) = \left(r + \gamma \max_{a_t} Q(s'_t, a'_t, \theta^-) - Q(s, a, \theta) \right)^2 \quad (11)$$

Here θ denotes the network weights, θ^- denote the target network weights and Q denotes the output of the deep learning network. As concurrent learning without communication results in poor learning, the coordinated learning with communication is proposed. Similarly, the parameter sharing scheme has limitations in bandwidth allocation and lesser coordination between the multi-agents. To ensure the coordination, the learning process is initiated with centralized learner.

Fig.2 shows the architecture of the proposed TFCMA-DRL.

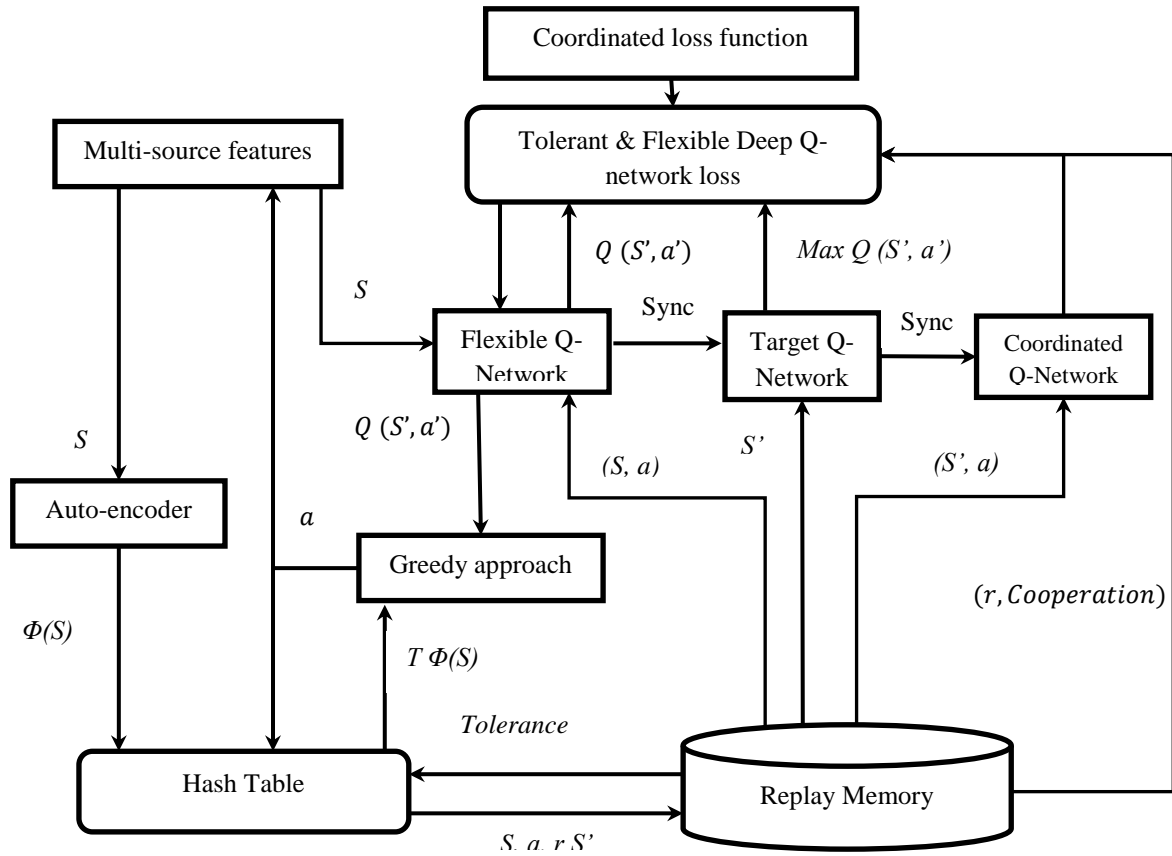


Figure. 2 Architecture of TFCMA-DRL

It computes the Q-learning gradient to update the Q-network and enables the network to share same weights between all agents. The tolerance is added to the flexible policy by

$$tol(s, t) = 1 - e^{-K \times Temp(\phi(s), a)} \quad (12)$$

Here $(\phi(s), a)$ is symbolized to the heat value $Temp$ and the hash-keys $\phi(s)$ are measured at time t for the state-action pair (s, a) .

The traditional methods to ensure coordination through communication between the multi-agents were equipped with the constraint optimization modelling which causes higher communication bandwidth and increased latency between the agent selections. To alleviate these limitations, the coordinated communication in TFCMA-DRL uses a small amount of bandwidth and reduced the computations for selecting the agents in the groups. This is achieved using joint action with communication parameter a^*

$$a^* = (a_i, a_g) \quad (13)$$

Here a_i and a_g are the action taken by individual agents and communicated actions between the agents, respectively. The communication from the nearer agents are joined by the agents own observation x_{it} from the state space as

$$s_{it} = x_{it}, a_{ft} \forall f, f \neq i \quad (14)$$

This model allows performing coordinated communication between the agents and allows the parameter sharing scheme to share the states and actions. However, one limitation in this model is that the action space gradually increases to reach $|a_i| \times |a_g|$. This reduces the effectiveness and hence discretion is used to select the group action. It is achieved through a modified joint action

$$a^* = \arg \max_{a_t} Q(s_t, a_t) \quad (15)$$

Additionally, the mini-batches and relay buffer are employed to improve the learning process and achieve stationary dynamics. The reward for each agent is allocated equally through appropriate credits. The difference reward is computed as

$$DR_i(z) = G(z) - G(z_i + u_i) \quad (16)$$

Here $G(z)$ is the global reward of unified state z and z_i and u_i are the system state without agent i and counterfactual term, respectively. As it is non-trivial to calculate the value of $G(z_i + u_i)$, the reward allocation is determined using another simple strategy formulated as

$$R_i = R \cdot Q_i(s_t, a_t) \left(\frac{1}{N} \sum_{i=1}^N Q_i(s_{it}, a_{it}) \right) \quad (17)$$

Here N denotes the number of agents, R is the global reward signal and $Q_i(s_{it}, a_{it})$ denote the state-action pair for agent i at time t . The lesser the value of $Q_i(s_{it}, a_{it})$, the better performance is achieved. Algorithm 1 summarizes the TFCMA-DRL model for stock forecasting.

Algorithm 1. TFCMA-DRL prediction model

Input: Tweets, news and stock quantitative data
 Begin
 Perform pre-processing
 Perform RBM
For each RBM model
 Initialize layers and state
 Compute energy and data passing
 Apply sentence2vec
End for
 Return event features
 Perform ELM for sentiment extraction
 Set the initial parameters
For each hidden layer
 Construct the network for training
 Compute output matrix using Eq. (2)
 Predict the sentiment class of each tweet
End for
 Perform ELM testing
 Return sentiment features
 Perform TFCMA-DRL
 Set learning rate α and β , N , weight θ , Maximum iterations, Initial $Temp$, t
 $t \leftarrow 0$
 Set all agents
For iteration $iter = 0, iter_max$ **do**
 Model objective function using Eq. (11)
 Apply tolerance policy using Eq. (12)
 Perform joint action using Eq. (15)
 Compute reward using Eq. (17)
Repeat
 Form relationship map
 Update weights for all agents
 Return gradients and loss
End

The implementation of this coordination policy will result in better learning of the multi-source sentiments and events. The TFCMA-DRL incorporates the stock indexes in a deep learning process and through the tolerant and coordination policies, achieves the high accuracy in stock forecasting.

4. Experimental Results

MATLAB tool is used to evaluate the proposed TFCMA-DRL model on the collected multi-source data. The proposed model is compared with algorithms namely CNN [17], MIL [21], TMA-DRL [22] and TFMA-DRL [23]. The proposed model as well as existing models are implemented and evaluated on the multi-source data that are described in Section 3.1. The existing classification algorithms are discussed in the literature studies for the common problem of stock market prediction and we have implemented them based on the author-defined features provided in their respective studies. The parameter setting for each of these algorithms is done by using the original author defined functions. The performance metrics namely accuracy, precision, recall, F1-score and execution time are used for evaluation.

Fig 3 shows the accuracy comparison for stock forecasting between the existing models and the proposed TFCMA-DRL. The comparison results justifies that the proposed TFCMA-DRL has higher accuracy than the existing models. TFCMA-DRL has accuracy of 96.667% which is 6.67%, 16.89%, 24.12% and 8.12% superior to TFMA-DRL, TMA-DRL, MIL and CNN models, respectively. The accuracy improvement of the proposed TFCMA-DRL is credited to its precise class distinguishing property with the signified features extraction through RBM and ELM.

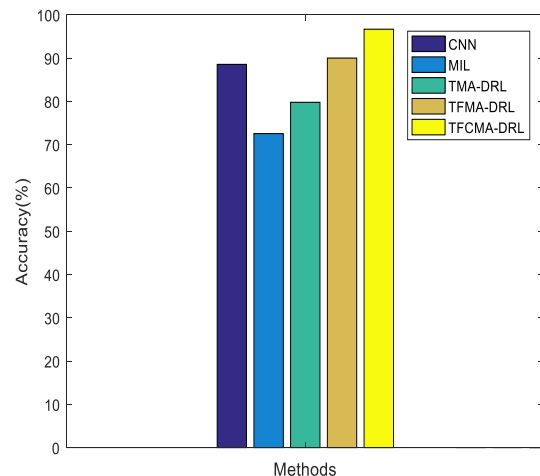


Figure. 3 Accuracy comparison

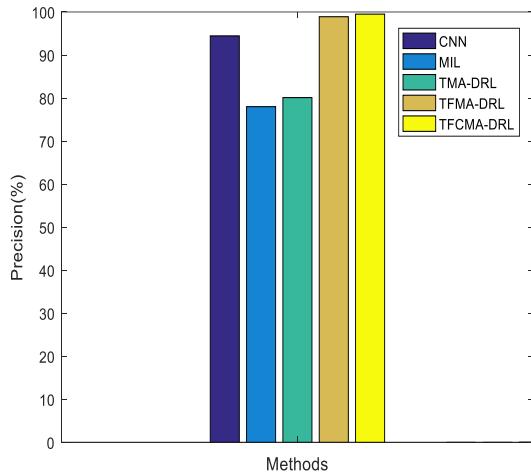


Figure. 4 Precision comparison

Fig.4 shows the precision comparison for stock forecasting between the existing models and the proposed TFCMA-DRL. Results proved that the TFCMA-DRL has high precision of 99.5% which is 0.6%, 19.4%, 21.5% and 5.1% superior to TFMA-DRL, TMA-DRL, MIL and CNN models, respectively. The precision values are better in TFCMA-DRL because of the effective extraction of coordinated class features.

Fig. 5 shows the recall comparison for stock forecasting between the existing models and the proposed TFCMA-DRL. The results show that the TFCMA-DRL has high recall than other models. It has recall of 96.67% and it is 6.67%, 16.07%, 23.97% and 6.34% superior to TFMA-DRL, TMA-DRL, MIL and CNN models, respectively. The recall values are better in TFCMA-DRL due to the accurate class distinguishing property and effective linking of the stock values towards the sentiment and event features.

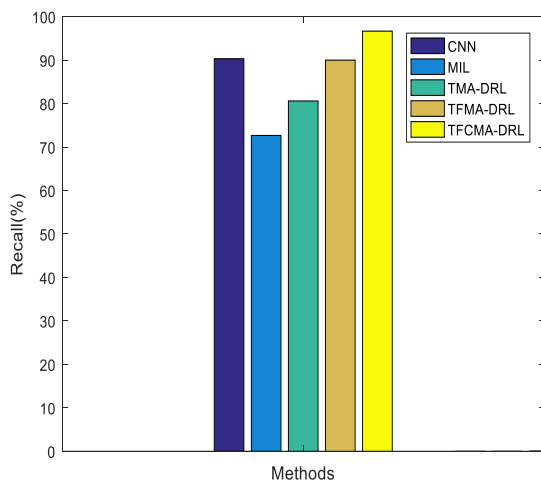


Figure. 5 Recall comparison

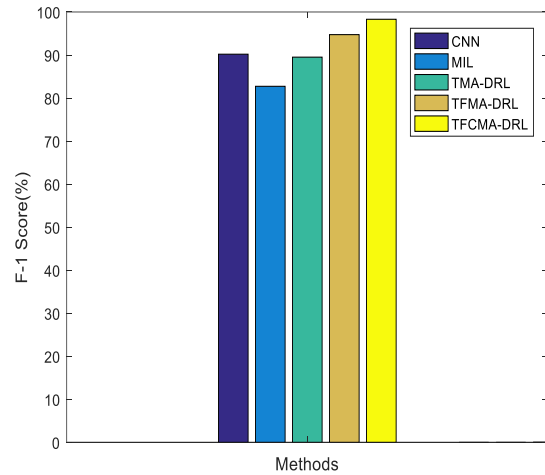


Figure. 6 F1-score comparison

Fig. 6 illustrates the f1-score evaluation for stock forecasting between the existing models and the proposed TFCMA-DRL. The plot results show that the TFCMA-DRL has high f1-score than other stock forecasting models. It has f1-score of 98.305% which is 3.57%, 8.77%, 15.57% and 8.11% superior to TFMA-DRL, TMA-DRL, MIL and CNN models, respectively. The F1-score values are high in TFCMA-DRL as the deep multi-agent processing of the data has reduced the error rate.

Fig. 7 illustrates the processing time evaluation for stock forecasting between the existing models and the proposed TFCMA-DRL. The proposed TFCMA-DRL is slightly complex due to the introduction of three different policies in deep reinforcement learning while also employ ELM and RBM models for stock indicators extraction. These complex models will increase the time complexity for training but due to fast training of these models, the time consumption is greatly minimized.

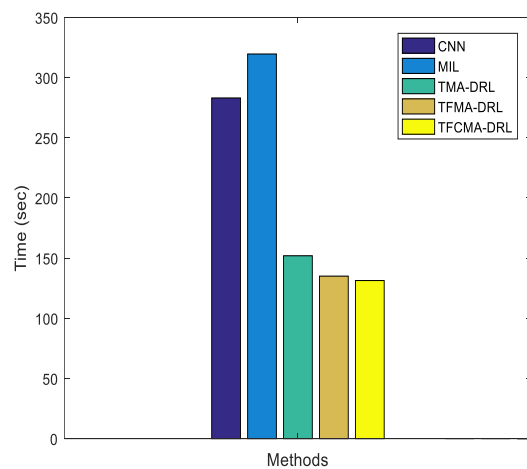


Figure. 7 Time comparison

From the figure, it is proved that the time complexity of the TFCMA-DRL with 131.357 seconds which is 3.65seconds, 19.6seconds, 188.23seconds and 152.3seconds less than TFMA-DRL, TMA-DRL, MIL and CNN models, respectively. The determined contribution for this efficiency is attributed to the use of superior design approach of TFCMA-DRL that minimizes the training time.

The CNN [17] model has the ability to learn deep features effectively but it has limitation of extensive design process which consumes more time during training. Compared to this CNN model, the proposed TFCMA-DRL has simple design process and hence takes less training time. The MIL [21] is similar to the CNN but it supports multi-instance feature learning from multi-source. But it lacks the knowledge based analysis to differentiate the precise classes. TFCMA-DRL model overcomes this limitation through its coordinated policy based learning agents. TMA-DRL [22] and TFMA-DRL [23] are stock prediction models that utilize the tolerance policy and flexibility policy, respectively, of the multi-agent deep learning. Through these policies, both these models learn the deep features from tweets, news and qualitative data. However, the complete learning of the features is still limited as their architecture lacks the relation mapping cooperation. The proposed TFCMA-DRL overcomes this problem through the relation mapping through coordinated policy. Due to these properties of the TFCMA-DRL, it has better performance than the other models.

5. Conclusion and Future work

An efficient prediction model for stock market price fluctuations is presented in this paper using TFCMA-DRL algorithm. This proposed model utilizes the cohesive sentiments and events extracted using ELM and RBM, respectively for the improvement in learning the price patterns. It is found that the proposed TFCMA-DRL model enhances the extraction of cohesive stock indicators and increases the accuracy of prediction without increasing the time complexity. The results highlighted that the TFCMA-DRL model achieved 96.67% accuracy, 99.5% precision, 96.67% recall and 98.305% F1-score values which are considerably greater than the compared models. Also, the TFCMA-DRL has less processing time of 131.357 seconds.

The future research directions include the possibilities of incorporating additional indicators by learning the profit margins, revenue reports and

philanthropy services of the companies. Additionally, the time complexity of the model can be further reduced if different policies are adaptively selected for the learning model.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

This work is a contribution of both the authors: Conceptualization, Chinnasamy Bhuvaneshwari and Raman Beena; methodology, Chinnasamy Bhuvaneshwari and Raman Beena; software, Chinnasamy Bhuvaneshwari and Raman Beena; validation, Chinnasamy Bhuvaneshwari and Raman Beena; formal analysis, Chinnasamy Bhuvaneshwari and Raman Beena; writing—original draft preparation, Chinnasamy Bhuvaneshwari and Raman Beena; writing—review and editing, Chinnasamy Bhuvaneshwari and Raman Beena.

References

- [1] L. Guiso, P. Sapienza and L. Zingales, "Trusting the stock market", *The Journal of Finance*, Vol.63, No.6, pp.2557-2600, 2008.
- [2] M. Van Rooij, A. Lusardi and R. Alessie, "Financial literacy and stock market participation", *Journal of Financial Economics*, Vol.101, No.2, pp.449-472, 2011.
- [3] A. Hameed, W. Kang and S. Viswanathan, "Stock market declines and liquidity", *The Journal of finance*, Vol.65, No.1, pp.257-293, 2010.
- [4] A. Degutis and L. Novickytė, "The efficient market hypothesis: A critical review of literature and methodology", *Ekonomika*, Vol.93, No.1, pp.7-23, 2014.
- [5] A. H. Moghaddam, M. H. Moghaddam and M. Esfandyari, "Stock market index prediction using artificial neural network", *Journal of Economics, Finance and Administrative Science*, Vol.21, No.41, pp.89-93, 2016.
- [6] J. Pan, Y. Zhuang and S. Fong, "The impact of data normalization on stock market prediction: using SVM and technical indicators", In: *Proc. of International Conference on Soft Computing in Data Science*, Singapore, pp.72-88, 2016.
- [7] Q. Qin, Q. G. Wang, J. Li and S. S. Ge, "Linear and Nonlinear Trading Models with Gradient Boosted Random Forests and Application to Singapore Stock Market", *Journal of Intelligent Learning Systems and Applications*, Vol.5, No.1, pp.1-10, 2013.

- [8] X. Li, C. Wang, J. Dong, F. Wang, X. Deng and S. Zhu, "Improving stock market prediction by integrating both market news and stock prices", In: *Proc. of International Conference on Database and Expert Systems Applications*, Berlin, Heidelberg, pp.279-293, 2011.
- [9] R. Batra and S. M. Daudpota, "Integrating StockTwits with sentiment analysis for better prediction of stock price movement", In: *Proc. of International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, Austria, pp.1-5, 2018.
- [10] S. Dargan, M. Kumar, M. R. Ayyagari and G. Kumar, "A survey of deep learning and its applications: A new paradigm to machine learning", *Archives of Computational Methods in Engineering*, Vol.2019, No.1, pp.1-22, 2019.
- [11] N. X. Yang, X. B. Jin, T. L. Su and J. L. Kong, "Multisource Data Analysis for Stock Prediction", In: *Proc. of 10th International Conference on Modelling, Identification and Control (ICMIC)*, India, pp.1-6, 2018.
- [12] S. M. Idrees, M. A. Alam and P. Agarwal, "A prediction approach for stock market volatility based on time series data", *IEEE Access*, Vol.7, No.1, pp.17287-17298, 2019.
- [13] S. C. Nayak, B. B. Misra and H. S. Behera, "ACFLN: artificial chemical functional link network for prediction of stock market index", *Evolving Systems*, Vol.10, No.4, pp.567-592, 2019.
- [14] C. Tang, W. Zhu and X. Yu, "Deep Hierarchical Strategy Model for Multi-Source Driven Quantitative Investment", *IEEE Access*, Vol.7, No.1, pp.79331-79336, 2019.
- [15] S. K. Chandar, "Fusion model of wavelet transform and adaptive neuro fuzzy inference system for stock market prediction", *Journal of Ambient Intelligence and Humanized Computing*, Vol.153, No.1, pp.1-9, 2019.
- [16] Q. Xu, L. Wang, C. Jiang and Y. Liu, "A novel (U) MIDAS-SVR model with multi-source market sentiment for forecasting stock returns", *Neural Computing and Applications*, Vol.32, No.10, pp.5875-5888, 2020.
- [17] E. Hoseinzade and S. Haratizadeh, "CNNpred: CNN-based stock market prediction using a diverse set of variables", *Expert Systems with Applications*, Vol.129, No.1, pp.273-285, 2019.
- [18] H. Chung and K. S. Shin, "Genetic algorithm-optimized multi-channel convolutional neural network for stock market prediction", *Neural Computing and Applications*, Vol.32, No.12, pp.7897-7914, 2019.
- [19] M. Wen, P. Li, L. Zhang and Y. Chen, "Stock market trend prediction using high-order information of time series", *IEEE Access*, Vol. 7, No. 1, pp. 28299-28308, 2019.
- [20] X. Li, P. Wu and W. Wang, "Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong", *Information Processing and Management*, Vol. 57, No. 5, pp. 102212-102218, 2020.
- [21] X. Zhang, S. Qu, J. Huang, B. Fang and P. Yu, "Stock market prediction via multi-source multiple instance learning", *IEEE Access*, Vol. 6, No. 1, pp. 50720-50728, 2018.
- [22] C. Bhuvaneshwari and R. Beena, "Stock Market Forecasting from Multi-Source Data using Tolerance Based Multi-Agent Deep Reinforcement Learning", *International Journal of Engineering and Advanced Technology (IJEAT)*, Vol. 9, No. 3, pp. 3492-3499, 2020.
- [23] C. Bhuvaneshwari and R. Beena, "Targeted Sentiments and Hierarchical Events based Learning Model for Stock Market Forecasting from Multi-Source Data", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, Vol. 9, No. 6, pp. 1763-1770, 2020.
- [24] G. E. Hinton, "Boltzmann machine", *Scholarpedia*, Vol. 2, No. 5, p. 1668, 2007.
- [25] G. B. Huang, Q. Y. Zhu and C. K. Siew, "Extreme learning machine: theory and applications", *Neurocomputing*, Vol. 70, No. 1-3, pp. 489-501, 2006.
- [26] E. Pesce and G. Montana, "Improving coordination in small-scale multi-agent deep reinforcement learning through memory-driven communication", *Machine Learning*, Vol. 155, No. 1, pp. 1-21, 2020.