World Academy of Science, Engineering and Technology
International Journal of Economics and Management Engineering
Vol:15, No:3, 2021

Lineup Optimization Model of Basketball Players Based on the Prediction of Recursive Neural Networks

Wang Yichen, Haruka Yamashita

***Abstract*—**In recent years, in the field of sports, decision making such as member in the game and strategy of the game based on then analysis of the accumulated sports data are widely attempted. In fact, in the NBA basketball league where the world's highest level players gather, to win the games, teams analyze the data using various statistical techniques. However, it is difficult to analyze the game data for each play such as the ball tracking or motion of the players in the game, because the situation of the game changes rapidly, and the structure of the data should be complicated. Therefore, it is considered that the analysis method for real time game play data is proposed. In this research, we propose an analytical model for "determining the optimal lineup composition" using the real time play data, which is considered to be difficult for all coaches. In this study, because replacing the entire lineup is too complicated, and the actual question for the replacement of players is "whether or not the lineup should be changed", and "whether or not Small Ball lineup is adopted". Therefore, we propose an analytical model for the optimal player selection problem based on Small Ball lineups. In basketball, we can accumulate scoring data for each play, which indicates a player's contribution to the game, and the scoring data can be considered as a time series data. In order to compare the importance of players in different situations and lineups, we combine RNN (Recurrent Neural Network) model, which can analyze time series data, and NN (Neural Network) model, which can analyze the situation on the field, to build the prediction model of score. This model is capable to identify the current optimal lineup for different situations. In this research, we collected all the data of accumulated data of NBA from 2019-2020. Then we apply the method to the actual basketball play data to verify the reliability of the proposed model.

***Keywords*—**Recurrent Neural Network, players lineup, basketball data, decision making model.

I. Introduction

In recent years, the analysis of sports data has been used to develop strategies for many sports such as baseball and soccer. The National Basketball Association (NBA) is a professional basketball league in North America, which is the largest basketball league in the world with annual revenues of over $4 billion since the 1950s. The NBA is made up of 30 teams; the team with a superior performance will receive more attention and financial support. Therefore, every team has a team of specialized analysts to analyze the various data for winning the games. On the other hand, along with the changes in NBA league rules in recent years [1], it has led to the rise of Small Ball strategy (Small Ball is a style of play that prioritizes the use of smaller players in the lineup, in order to increase speed, agility, and scoring at the expense of the players' height and strength). On the other hand, recently, the Golden State Warriors set an NBA record for wins in the 2014-15 season and reached to finals in a seasonal league for five times and won three championships in five years. This led all teams to follow the strategy of Warriors "Small Ball". Now, many teams are making effort to improving the team's offensive scoring ability and optimizing the team's line up. Then, many teams are adopting Small Ball lineups for making advantages for their games. However, it is considered that there are situations that Small Ball lineup should be adopted and should not be adopted. For discussing the optimal lineup, it is required to predict how much the advantage for changing the lineup is; therefore, we need to predict the point of any lineup.

There are many statistical analysis using statistical model is proposed; for example, predict model for the player and ball movement [2], illustrating model of the important players in each game using graph structure [3], and the transition of the ball during the offense are proposed [4]. In this study, not only individual scoring prediction but also the whole lineup scoring prediction is performed. Recently, several studies using deep NN model [2], [5] for analyzing basketball data are proposed. Especially, basketball analysis based on the RNN model [5], which is the high-performance model for the time series data prediction, is derived [6], [7]. However, the RNN based studies have been forcing on prediction on plays such as the movement of players also the movement of ball. Therefore, constructing the prediction model of team should be required. We propose a method to predict by combining the NN and RNN. Moreover, we show how to optimize the lineup using the prediction model. However, for considering the entire lineup, the actual problem is usually whether the part of lineup needs to be changed or not. Moreover, the current main issue is whether to use a Small Ball lineup or not. This research shows how to realize the player optimization considering the partial position optimization and Small Ball lineups.

II. Preliminaries

A. Data

In this research, we collected data by watching the 2019-20 Los Angeles Lakers [8] games and recording the game conditions on each possession. Details are shown in Table I. The first 10 were used as explanatory variables to determine the current on-court situation, and the Score was used as the aim

Wang Yichen is with the Sophia University, Japan (e-mail: y-wang-9u9@eagle.sophia.ac.jp).

variable of this experiment.

TABLE I
ATTRIBUTES OF THE DATA

Attributes	Comment
Position	PG,SG,SF,PF,C every position player's number
Quarter	Which of the four quarters in the game
Time consuming	Length of time consumed by the offense
Score difference	Current score difference between the two teams
Team fouls	Current cumulative fouls for both teams
Winning/losing streaks	The posture of the team
Size comparison	Current Player Size Comparison
Top Defender	Whether the top defensive players on the field
Average points lost	Opponent's last season average points lost
Home/Away	Home and Away Relationship
Score	Score of this offense

B. RNN

RNN [9] is a framework that takes sequence data as input, recursion in the direction of sequence evolution, and all nodes are connected in a chain.

RNN is applied to the model for NBA play data and confirmed a significant accuracy rate [6], [7].

III. MODEL

In this study, for constructing a model enables to predict the score of the time greatly, we use the explanatory variables and model structure as follows.

Because basketball is a multiplayer competitive sport, being on a roll of (preferable situation) team is important for the team's offense. For grasping situation of the team, the 5 scores of the current plays are selected as explanatory variables, and the explanatory variables are included in the RNN model. Next, since this is a model for predicting score of the team which is scored by players on the field, and each player is identified by a corresponding jersey number, we accumulated the discrete data. Moreover, all teams in the league are eager to win, and the scoring efficiency of the team directly affects the team's victory or defeat. Thereby the lineup that can be more efficient in the ever-changing game is the most optimal lineup that can lead the team to victory. We added the explanatory variable of whether the situation is Small Ball or not.

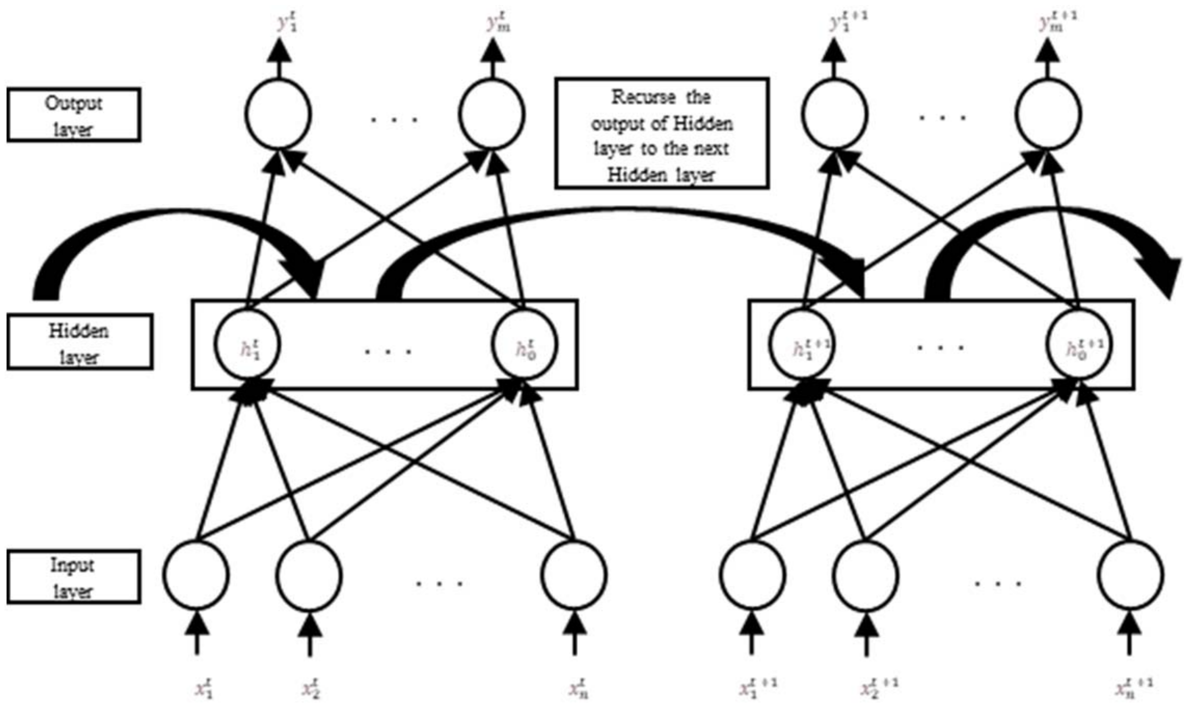

Fig. 1 The structure of RNN

A. Forward Propagation

RNN Model, at time t, let $\boldsymbol{x^t} = (x_1^t, \dots x_i^t, \dots x_n^t)$ be the input vector for a RNN model, consisting of the 5 offensive scores before time t, $\boldsymbol{h^t} = (h_1^t, \dots h_k^t, \dots h_o^t)$ be the output vector of the hidden layer, and $\boldsymbol{y^t} = (y_1^t, \dots y_j^t, \dots y_m^t)$ is the output vector, which are expressed as follows. Let $W^{(in)} = \{w_{ik}^{(in)}\}$, $W^{(out)} = \{w_{kj}^{(out)}\}$, $W = \{w_{k'k}\}$ be the weights of the branches connecting the input layer and the hidden layer, the weights of the branches connecting the hidden layer and the output layer, and the weights of the branches connecting the hidden layer at the previous time and the hidden layer at the current time, respectively. Then, h_k^t can be represented as:

$$h_k^t = f(u_k^t) = f(\textstyle\sum_{i=1}^{n} x_i^t \, w_{ik}^{(in)} + \sum_{k'=1}^{o} h_{k'}^{t-1} \, w_{k'k}) \quad (1)$$

where f is the activation function. In this way, the values of the intermediate layer at the previous time are weighted and input to the intermediate layer at the current time. Then, the output of y_j^t is calculated as (2):

$$y_j^t = g(v_j^t) = g(\sum_{k=1}^{o} h_k^t\, w_{kj}^{(out)}) \quad (2)$$

where g is the activation function.

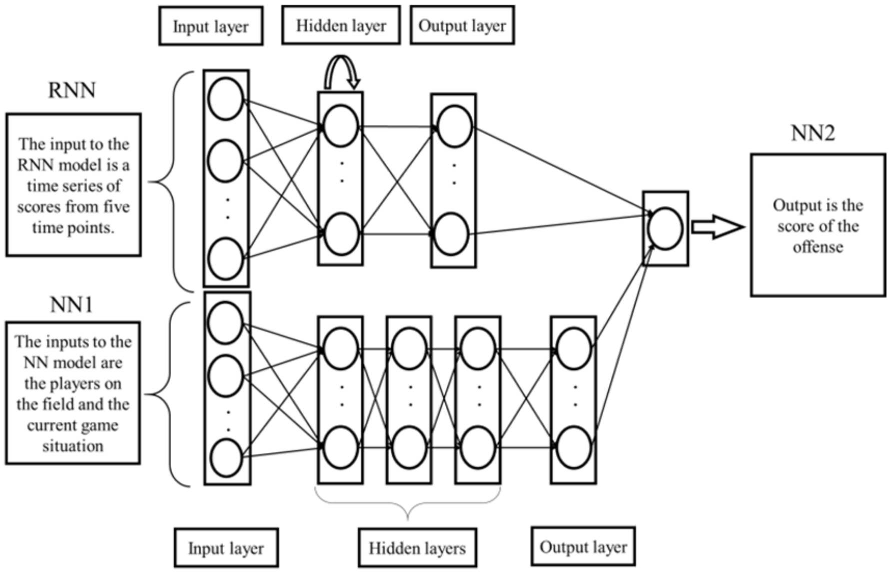

Fig. 2 The structure of our proposal model

In the NN1 model, we consider the multi hidden layered structure. First, l is the number of layers, $l = 1$ is the input layer, $l = 2,3,4$ is the hidden layers, and $l = 5$ is the output layer. Let $\boldsymbol{x^{l'}} = (x_1^{l'}, \dots x_n^{l'})$be the input to the NN1 model, consisted of the information of players, and the current game situation, respectively. Also, let $\boldsymbol{h^{l'}} = (h_1^{l'}, \dots h_n^{l'})$ be the hidden layer output, and $\boldsymbol{y^{l'}} = (y_1^{l'}, \dots y_n^{l'})$'is the output.

$$\boldsymbol{h^{(l+1)'}} = W^{(l+1)'}\boldsymbol{z^{(l)}} \quad (3)$$

$$\boldsymbol{z^{(l+1)}} = \boldsymbol{f'}(\boldsymbol{h^{(l+1)'}}) \quad (4)$$

$$\boldsymbol{y^{l'}} \equiv \boldsymbol{z^{(L)}} \quad (5)$$

$\boldsymbol{f'}$ is the activation function, $W^{(l+1)'}$is the weight to the next layer.

In the NN2 model the output $\boldsymbol{y^t}$ of the RNN model and the output $\boldsymbol{y^{l'}}$ of the NN model are used as inputs again to obtain the final score of the offense. P is the output vector of the NN1 output layer, Q is the output vector of the RNN output layer, and $Score$ is the output of this research which are expressed as follows. We denote the weight of layer between NN2 and RNN, and layer between NN2 and RNN by W'', W''', respectively.

$$P = g'(y^{l'}W'')$$
$$Q = g''(y^t W''')$$
$$Score = g'''(P + Q) \quad (6)$$

where g', g'', g''' is the activation function, respectively.

B. Backpropagation

In learning, we update the parameters W. Since the gradient method is used for updating, it is necessary to calculate the derivatives of these parameters with respect to the loss function. Let E be an arbitrary loss function, the derivative of each parameter can be calculated by (6)-(16).

NN2 Model

For minimizing E in terms of the layer between NN2 and RNN is as follows:

$$\frac{\partial \mathrm{E}}{\partial w''} = \sum_{n=1}^{n} \frac{\partial \mathrm{E}}{\partial P} \frac{\partial P}{\partial w''} = \sum_{n=1}^{n} \frac{\partial \mathrm{E}}{\partial P} \frac{\partial}{\partial w''}\left(\sum_{n=1}^{n} \boldsymbol{y^{l'}} W''\right) = \sum_{n=1}^{n} \frac{\partial \mathrm{E}}{\partial P}\, \boldsymbol{y^{l'}} (7)$$

For minimizing E in terms of the layer between NN2 and NN1 is as:

$$\frac{\partial \mathrm{E}}{\partial w'''} = \sum_{t=1}^{T} \frac{\partial \mathrm{E}}{\partial Q} \frac{\partial Q}{\partial w'''} = \sum_{t=1}^{T} \frac{\partial \mathrm{E}}{\partial Q} \frac{\partial}{\partial w'''}\left(\sum_{j=1}^{m} y^t\, w'''\right) = \sum_{t-1}^{T} \frac{\partial \mathrm{E}}{\partial Q}\, y^t \quad (8)$$

C. RNN Model

Using (8), optimization of $w_{ik}^{(in)}, w_{k'k}, w_{kj}^{(out)}$ respectively such that minimizes E is represented as follows:

$$\frac{\partial \mathrm{E}}{\partial w_{ik}^{(in)}} = \sum_{t=1}^{T} \frac{\partial \mathrm{E}}{\partial Q} \frac{\partial Q}{\partial w'''} \frac{\partial \mathrm{w'''}}{\partial v_j^t} \frac{\partial v_j^t}{\partial w_{kj}^{(out)}} \frac{\partial w_{kj}^{(out)}}{\partial u_k^t} \frac{\partial u_k^t}{\partial w_{ik}^{(in)}}$$
$$= \sum_{t=1}^{T} \frac{\partial \mathrm{E}}{\partial Q}\, y^t\, \frac{\partial}{\partial w_{ik}^{(in)}}\left(\sum_{i=1}^{n} x_t^t\, w_{ik}^{(in)} + \sum_{k'=1}^{o} h_{k'}^{t-1}\, w_{k'k}\right) (9)$$

$$\frac{\partial \mathrm{E}}{\partial w_{k'k}} = \sum_{t=1}^{T} \frac{\partial \mathrm{E}}{\partial Q} \frac{\partial Q}{\partial w'''} \frac{\partial \mathrm{w'''}}{\partial v_j^t} \frac{\partial v_j^t}{\partial w_{kj}^{(out)}} \frac{\partial w_{kj}^{(out)}}{\partial u_k^t} \frac{\partial u_k^t}{\partial w_{k'k}}$$
$$= \sum_{t=1}^{T} \frac{\partial \mathrm{E}}{\partial Q}\, y^t\, \frac{\partial}{\partial w_{k'k}}\left(\sum_{i=1}^{n} x_t^t\, w_{ik}^{(in)} + \sum_{k'=1}^{o} h_{k'}^{t-1}\, w_{k'k}\right) (10)$$

$$\frac{\partial \mathrm{E}}{\partial w_{kj}^{(out)}} = \sum_{t=1}^{T} \frac{\partial \mathrm{E}}{\partial Q} \frac{\partial Q}{\partial w'''} \frac{\partial \mathrm{w'''}}{\partial v_j^t} \frac{\partial v_j^t}{\partial w_{kj}^{(out)}}$$
$$= \sum_{t=1}^{T} \frac{\partial \mathrm{E}}{\partial Q}\, y^t\, \frac{\partial}{\partial w_{kj}^{(out)}}\left(\sum_{k=1}^{o} h_k^t\, w_{kj}^{(out)}\right) \quad (11)$$

$$\frac{\partial w'''}{\partial u_k^t} = \sum_{j=1}^{m} \frac{\partial w'''}{\partial v_j^t}\frac{\partial v_j^t}{\partial u_k^t} + \sum_{k'=1}^{o} \frac{\partial w'''}{\partial u_{k'}^{t+1}}\frac{\partial u_{k'}^{t+1}}{\partial u_k^t} \tag{12}$$

By setting $\frac{\partial w'''}{\partial u_k^t} = \delta_k^t$, (12) can be transformed as:

$$\delta_k^t = \{\sum_{j=1}^{m} \frac{\partial w'''}{\partial v_j^t}\, w_{kj}^{(out)} + \sum_{k'=1}^{o} \delta_{k'}^{t+1}\, w_{k'k}\} f'(u_k^t) \tag{13}$$

NN1 Model

Using (7), optimization of $W^{l'}$ respectively such that minimizes E is represented as follows:

$$\frac{\partial E}{\partial w^l} = \frac{\partial E}{\partial P}\frac{\partial P}{\partial w''}\frac{\partial w''}{\partial h^l}\frac{\partial h^l}{\partial z^l} \tag{14}$$

$$\frac{\partial E}{\partial h^l} = \sum_{n=1}^{n} \frac{\partial w^{(l+1)}}{\partial \boldsymbol{h}^{(l+1)}}\frac{\partial \boldsymbol{h}^{(l+1)}}{\partial w^l} \tag{15}$$

By setting $\frac{\partial w''}{\partial h^l} = \delta^{(l)}$, (15) can be transformed as:

$$\delta^{(l)'} = \sum_{n=1}^{n} \delta^{(l+1)}\, (w^{(l+1)} f'(h^{l'})) \tag{16}$$

D. Model Evaluation

Nine games from the 2019-20 season were used as training data, and the tenth game was used as test data for the model to make scoring predictions as a group of every ten offensive possessions.

x_t is the predicted score for each possession, y_t is the actual score for each offense.

$$x_t = (x_1, \ldots x_t)$$
$$y_t = (y_1, \ldots y_t)$$
$$Sum\ of\ the\ Prediction = \sum_1^T x_t$$
$$Sum\ of\ the\ Observation = \sum_1^T y_t$$

The average R^2 is 0.5533 and the average correlation coefficient is 0.7799. Then we determined the activation of these data.

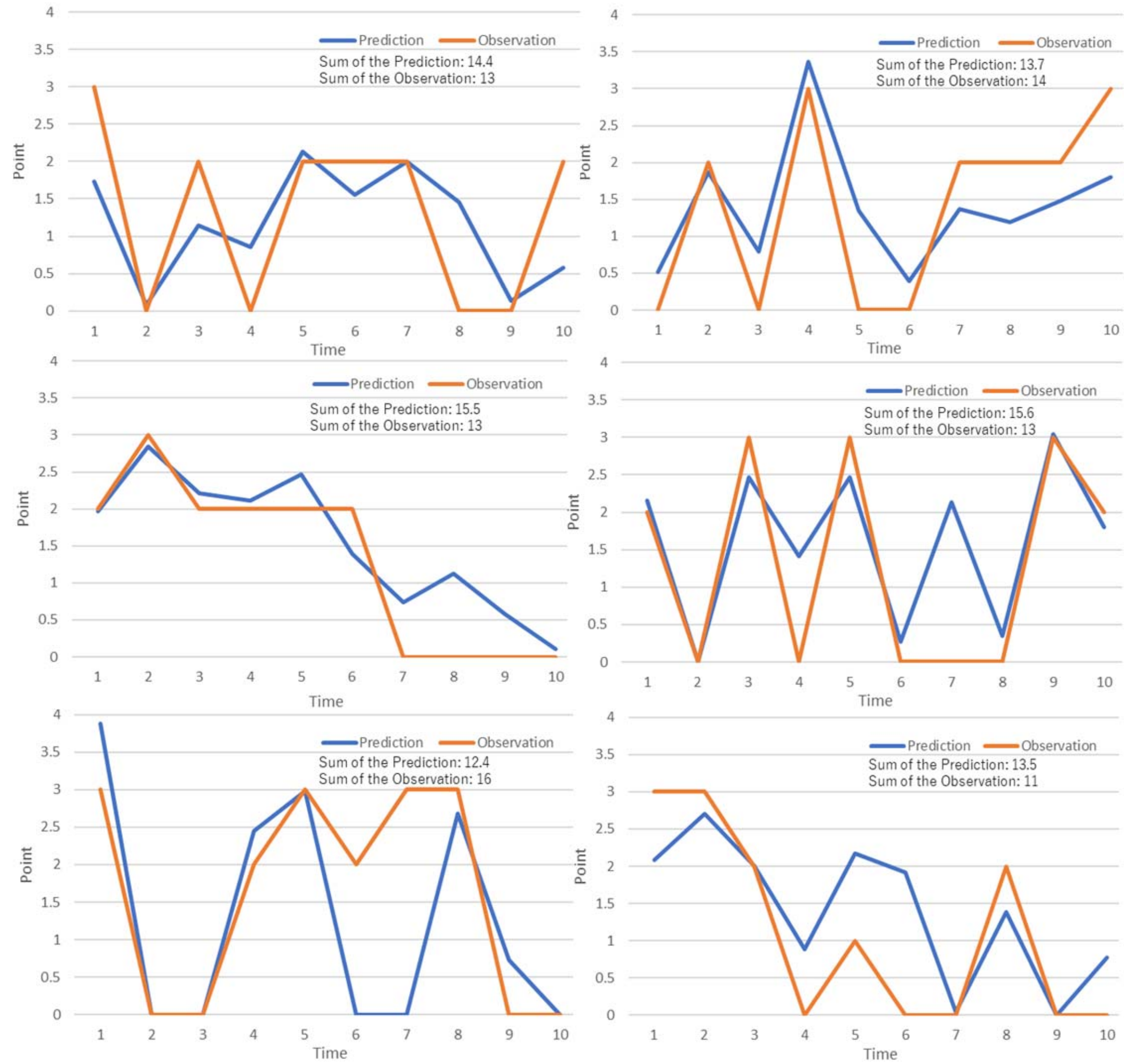

Fig. 3 Results of the test data

IV. Analysis of the Real Data

A. Hyperparameter Setting

The explanatory variables are the formation of the team members for each play, the current flow of the game, the opponent's defense, and the time series data of past scores for five time points. The objective variable is the current score. Each hidden layer has 3 layers, 16 nodes for NN, 32 nodes for RNN. We used a combination of three activation functions: Sigmoid, tanh (hyperbolic tangent), and ReLu (ramp function). epoch is set to 40 times.

B. Experiments Setting

Optimizing Players by Focusing on Small Ball

Lakers now have two main lineups made up of their strongest players, the normal (starting) lineup and the "Small Ball" lineup; however, four players of these two lineups are overlapping. Therefore, the remaining one is optimized to determine the members. The game situation data were randomly taken 10 times for the prediction of the scoring of the two lineups, and only one position player was changed in each experiment while controlling for the same 4 players in Fig. 4.

There are 4 candidates at the Point Guard (PG) position and 3 post selections at the Center (C) position depending on the position of the player and the reasonableness of the lineup.

Predicting Line-Up Performance

In the game, a coach has a crucial responsibility to manage the team to get the best result out of the game. Showing an efficient and dynamic strategy to use all the resources of a team for succeeding in a matchup defines the outcome of a game [8] is required. Moreover, in the basketball games, score changes dramatically on the court. Moreover, the Lakers have two powerful lineups; Small Ball lineup and Normal lineup. Therefore, in this experiment, for different game situations, the performance of the two lineups is predicted. Moreover, for understanding the performance, we summarize the prediction results in terms of four situations: Lead, Behind, Contend, and Big lead.

C. Results of Experiment

Results of Optimizing Players by Focusing on Small Ball

Fig. 5 represents the experiment results for the optimization whether small ball strategy should be adopted or not.

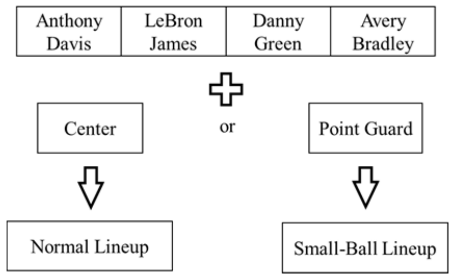

Fig. 4 Composition of the two lineups

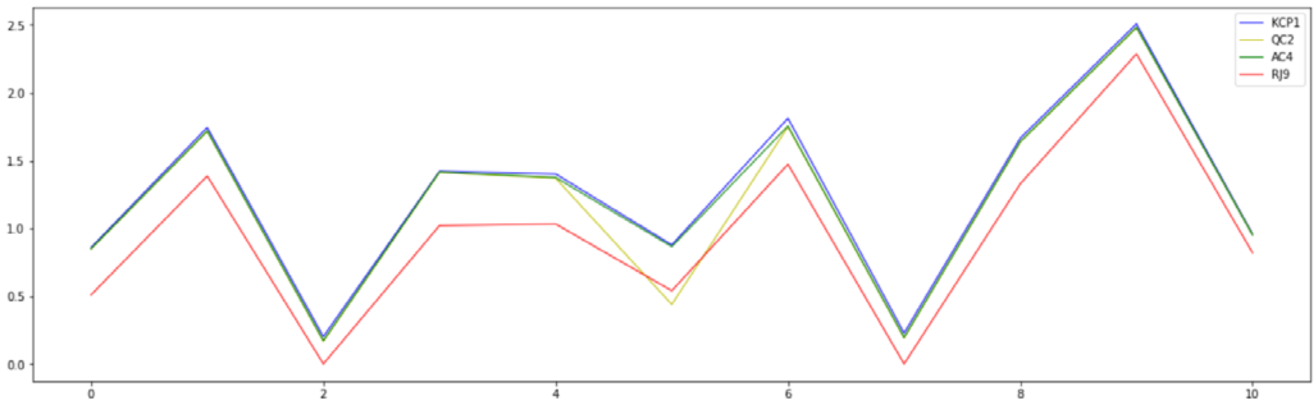

Fig. 5 Result of the prediction for PG position (vertical axis represents the score and horizontal axis represents time)

TABLE II
Result of the Prediction for PG Position

Name	Total Score
Kentavious Caldwell-Pope	13.67
Quinn Cook	12.96
Alex Caruso	13.42
Rajon Rondo	10.39

TABLE III
Result of the Prediction for C Position

Name	Total Score
JaVale McGee	10.21
Dwight Howard	14.08
Devontae Cacok	13.48

According to Fig. 5, KCP (Kentavious Caldwell-Pope) has the highest scoring efficiency; he is the most suitable player at the PG position in the Small Ball lineup.

Fig. 6, corresponding to the result of prediction for C position, shows that DH (Dwight Howard) has the highest scoring efficiency; he is the most suitable player at the C position in the Normal lineup. Once we found the best players for the two lineups, we compared the scoring of the two lineups again.

As a result, the total score for the small ball lineup was 11.3, exceeded three times, the total score for the normal lineup was 12.6, exceeded five times, and the score was roughly the same three times.

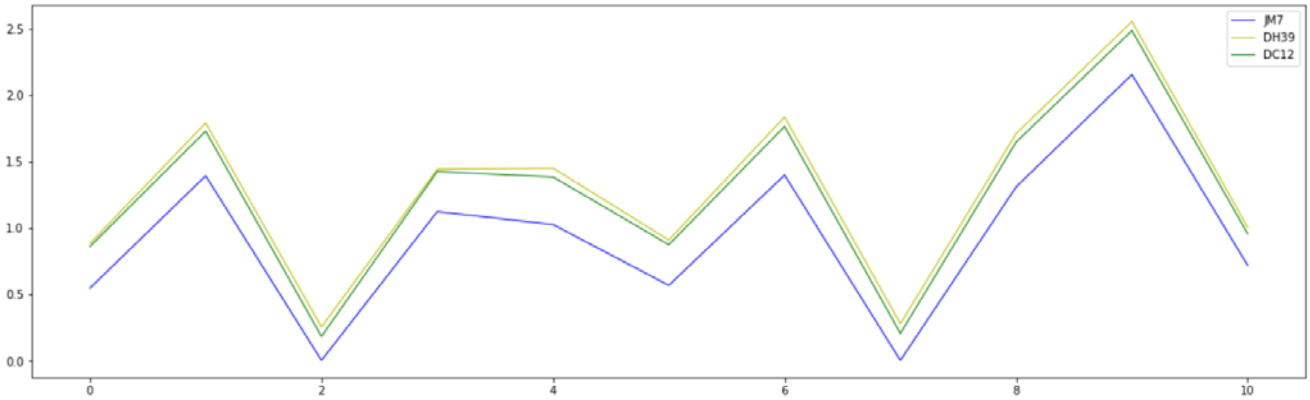

Fig. 6 Result of the prediction for C position (vertical axis represents the score and horizontal axis represents time)

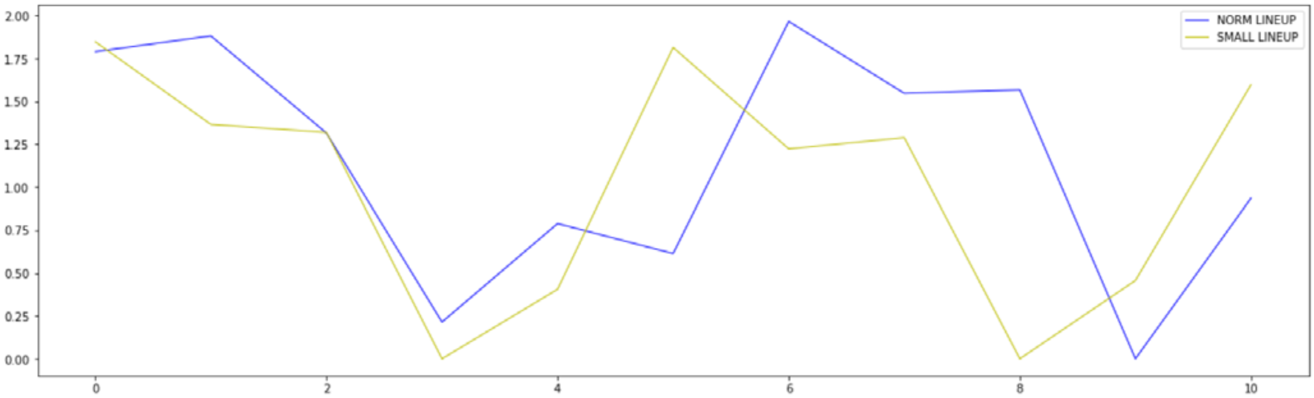

Fig. 7 Result of the prediction for two lineups (vertical axis represents the score and horizontal axis represents time)

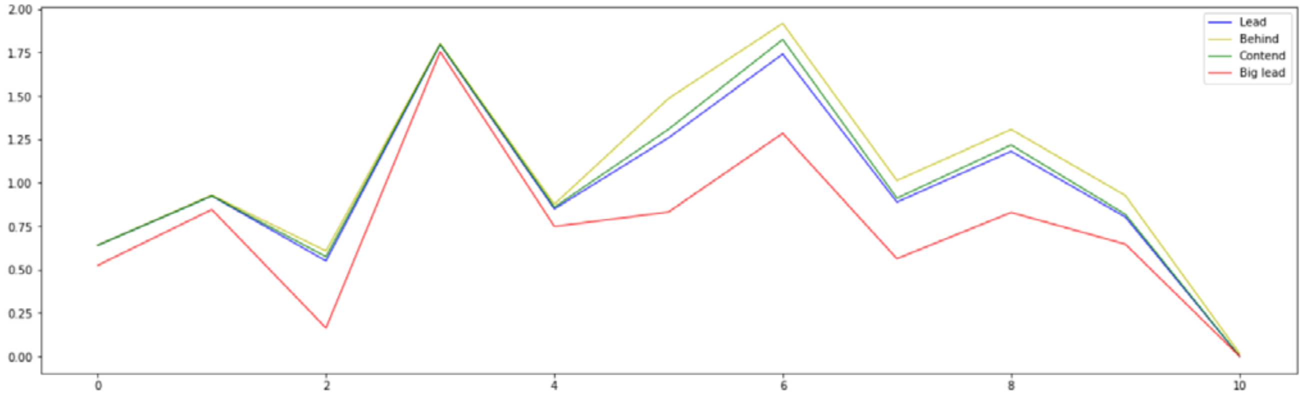

Fig. 8 Result of the prediction for small-ball lineup (vertical axis represents the score and horizontal axis represents time)

Results of Predicting Lineup Performance

TABLE IV
RESULT OF SMALL-BALL LINEUP

Situation	Total Score
Lead	10.61
Behind	11.51
Contend	10.86
Big lead	8.18

The results show that the two lineups seem to score best when the team is losing, but the Small Ball lineup performs better when the team is leading, and the Normal lineup performs better when the team is Big lead.

TABLE V
RESULT OF NORMAL LINEUP

Situation	Total Score
Lead	10.45
Behind	11.39
Contend	10.87
Big lead	8.29

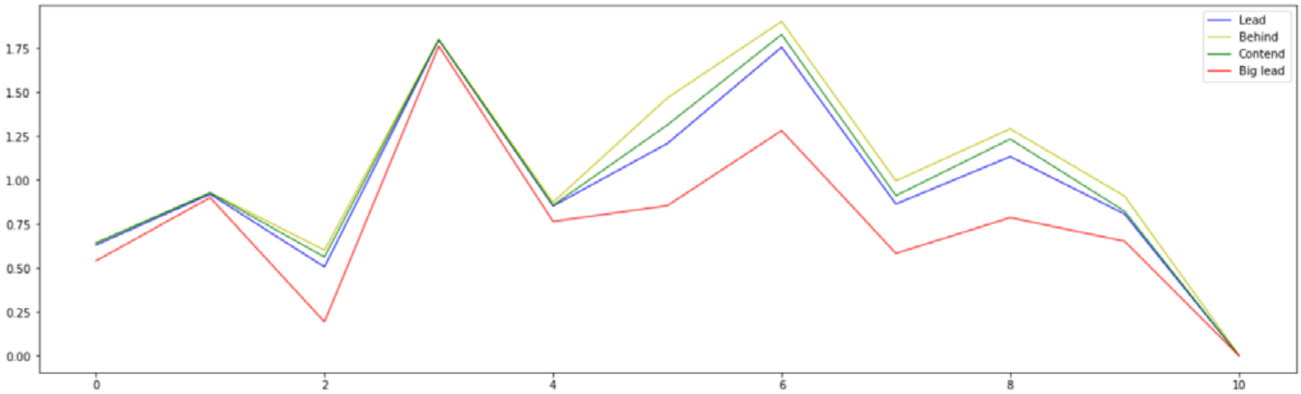

Fig. 9 Result of the prediction for small-ball lineup (vertical axis represents the score and horizontal axis represents time)

D. Examination

Applicability of Small Ball: Although the importance of the SMALL BALL lineup is recognized by all teams in today's basketball, this experimental analysis shows that not all teams fit into the SMALL BALL lineup and that in some situations it is easier to score when the normal lineup is used. It was found that they were more likely to score points.

Team performance in match situations: We found that regardless of the lineup, the scoring efficiency is higher when the team is trailing instead of decreasing when the team is leading, and the scoring efficiency decreases even more severely when the team is leading by a large number of points.

V. Conclusion

In this study, a combination of RNN and NN was used to construct a prediction model with good accuracy. The model combines not only the on-court situation but also the time-series scoring data to achieve the scoring prediction for lineups. Then we find the player with the highest scoring efficiency through each player's socring situation and complete the lineup optimization. This research can be applied not only in basketball, but also in soccer and volleyball, etc. It can be applied in all different sports which have a long-term strategy in a game, affected by many factors during the competition or substitutions. However, we are currently modeling and predicting for only one team, and the challenge is to show the effectiveness of the model for multiple teams.

References

[1] Nourayi, Mahmoud M. "Strategically Driven Rule Changes in NBA: Causes and Consequences." *The Sport Journal* 22 (2019), published online.

[2] Wang, Kuan-Chieh, and Richard Zemel. "Classifying NBA offensive plays using neural networks." *Proceedings of MIT Sloan Sports Analytics Conference. Vol. 4. 2016*, published online.

[3] Piette, James, S. Anand, and L. Pham. "Evaluating basketball player performance via statistical network modeling." *The 5th MIT Sloan Sports Analytics Conference*. 2011, published online.

[4] Clemente, Filipe Manuel, et al. "Network analysis in basketball: Inspecting the prominent players using centrality metrics." *Journal of Physical Education and Sport* 15.2 (2015): 212, published online.

[5] Yin, Wenpeng, et al. "Comparative research of CNN and RNN for natural language processing." *arXiv preprint arXiv:1702.01923* (2017), published online.

[6] Wang, Kuan-Chieh, and Richard Zemel. "Classifying NBA offensive plays using neural networks." *Proceedings of MIT Sloan Sports Analytics Conference. Vol. 4. 2016*, published online.

[7] Shah, Rajiv, and Rob Romijnders. "Applying deep learning to basketball trajectories." *arXiv preprint arXiv:1608.03793* (2016), published online.

[8] https://www.nba.com/lakers/

[9] Mikolov, T., Kombrink, S., Burget, L., Černocký, J., & Khudanpur, S. , Extensions of recurrent neural network language model. In: 2011 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, pp. 5528-5531, (2011).