Basketball Scouting Framework Considering Interaction among Players

Haruhiko Tashiro,* Mao Nishiguchi, Takuya Shimano, Kenta Morikawa, Ryota Nagano, Fujio Toriumi

***Abstract*—Finding and acquiring promising talent has always been one of the most important management issues in various organizations. In professional sports, this activity is referred to as scouting. For a long time, it was left to the intuition and experience of individuals. To make scouting more rational, it is important to quantitatively evaluate the skills of potential players. Traditional methods of evaluating player ability have focused on assessing the individual, but another approach is possible that evaluates player ability while considering compatibility with other players. In this study, we developed a sports player ability estimation method for basketball that considers compatibility between players by using Factorization Machines, which are machine learning models mainly used in recommendation systems and known for their superiority in extracting interactions between elements. In addition, we proposed a player scouting framework based on our developed method. Experiments based on professional basketball leagues showed that the proposed framework can estimate abilities more realistically than existing methods and has effective properties for scouting players. The widespread use of the system based on the proposed framework is expected to improve the efficiency of scouting, increase the liquidity of the player market, and reduce the mismatch between teams and players, thereby increasing the level of competition and revitalizing the professional sports industry.**

***Index Terms*—Sports Analytics, Factorization Machines, Basketball**

I. INTRODUCTION

In sports team management, the winning percentage or performance of a team is one of the most important performance indicators. Several studies indicate a relationship between the winning percentage and the management situation/profit of teams [1, 2]. Therefore, general managers must strive to build a system that improves performance to maximize the win rate, as well as identify new influential players. Thus, as in any organization, one of the most important management decisions is to extend invitations to join a team. However, in sports, this framework, known as scouting, is typically based on the intuition and experience of individuals. Consequently, players join a team with high expectations but often perform below expectations, which is a serious management issue. Therefore, quantitative decision-making is required to make scouting more rational. To achieve this objective, particularly in team sports, we believe that a quantitative evaluation framework that satisfies the following three conditions is required.

- Ability to evaluate performance as an individual.
- Ability to evaluate performance as a lineup.
- Ability to predict unknown lineup performance.

To satisfy the first condition, which focuses on the evaluation of individual abilities, the plus/minus rating method is typically employed [3]. This method expresses the number of points scored by each player by assigning the number in the field. The most significant advantage of this method is that it can express various abilities, including the offensive or defensive abilities of players, in a single dimension, i.e., scoring. The simplicity of this method renders it applicable to various sports teams. Different application methods based on this method have been proposed in basketball, i.e., the adjusted plus/minus (APM) method [4], which uses linear regression to assign scores, and the regularized APM (RAPM) [5], which uses ridge regression for the same purpose.

However, in team sports, lineup abilities cannot always be expressed as the sum of each player's ability in lineups. In other words, players' personal chemistry and role assignments are influencing factors. Therefore, appropriate methods must be adopted to evaluate and estimate performance as a lineup, which cannot be measured by individual abilities. In this regard, a method known as "extended RAPM", which is a version of RAPM that enables lineup evaluation as well as individual evaluation, has demonstrated relatively good performance in experiments for estimating known lineup evaluations [6]. However, this method [6] cannot completely reflect interactions in the model and effectively predict the lineup performance when an unknown player joins the team. To the best of our knowledge, studies regarding the estimation of unknown lineups in the basketball domain have not been conducted.

In this study, to develop a scouting support system for basketball, we propose a framework that makes player recommendations based on lineup performance estimation by considering interactions among players. An overview of the proposed system is shown in Fig. 1. The system obtains data from various leagues worldwide and uses this data to establish a performance estimation model and implement the appropriate filtering logic. The general manager receives an ordered list of suitable players for the team. The proposed framework corresponds to the performance estimation and

Manuscript received February 10, 2023; revised April 10, 2023.

H. Tashiro is with Graduate School of Engineering, The University of Tokyo, Japan (e-mail: tashiro@torilab.net).

M. Nishiguchi is with Graduate School of Engineering, The University of Tokyo, Japan (e-mail: nishiguchi.mao@mail.u-tokyo.ac.jp).

T. Shimano is with SportMeme, inc, Japan (e-mail: t.shimano.55@sportmeme.co).

K. Morikawa is with SportMeme, inc, Japan (e-mail: morikawa.kenta.mystery@gmail.com).

R. Nagano is with SportMeme, inc, Japan (e-mail: r.nagano.47@sportmeme.co).

F. Toriumi is with Graduate School of Engineering, The University of Tokyo, Japan (e-mail: tori@sys.t.u-tokyo.ac.jp).

*Correspondence: tashiro@torilab.net

filtering shown in Fig. 1.

Fig.1. Overall structure of our scouting support system.

The proposed method is based on the concept of Factorization Machines (FM) [7], which is a machine learning model originally developed for recommendation algorithms, such as those for click-through rate prediction. It is known for its ability to extract interactions between elements from sparse datasets, making it appropriate for this task, which attempts to explicitly incorporate interactions between players. Furthermore, it is applicable for estimating unknown interactions [7], which satisfies the requirements of the proposed method. Recently, several deep neural network (DNN) [8] models, extended from FM to manage sparse and dense mixed datasets, have been proposed [9], demonstrating the development of this field. In this study, we adopt xDeepFM [10], which is one of the most effective models among the existing models, as the main model to manage both sides of the dataset: the sparse dataset (player information that comprises each lineup) and the dense dataset (statistics for each player).

Using six years' worth of EuroCup [12] and EuroLeague [13] play-by-play (PbP) data and the statistical data of players (box score) obtained by scraping official websites, we conducted several comparative experiments to confirm the effectiveness of our proposed method. Firstly, the results demonstrate that our method performs better than previous methods. Furthermore, we show that the model can estimate the performance of unknown lineups with an accuracy comparable to that of known lineups. Additionally, we verify that interactions among players contribute to the accuracy of team performance estimation and that a certain number of players are predicted to perform well only on particular teams.

The remainder of this paper is organized as follows: Section II summarizes related studies, Section III describes the datasets used, Section IV explains the proposed method, Section V presents the experiments and their results, and Section VI provides the conclusions and recommendations for future studies.

II. Related Works

A. Plus/Minus Rating

The Plus/Minus rating [3] is a method for expressing each player's ability in terms of the number of points scored or lost based on the score difference during play. For example, if a player scores 30 points for their team and their team wins by 10 points, then the player's Plus/Minus rating (PM) is +20. Conversely, if the player scores 20 points but their team loses by 10 points, their PM is −10.

This method is based solely on the score and is independent of the specific sport. It is a top-down method used in many team sports, including basketball, ice hockey, and soccer. However, due to its design, players who are on the same team share the same PM score and are significantly affected by other players. For example, if an extremely skillful player is assigned to a team, they will enhance the values of the other players, and vice versa. The typical methods for solving this problem are the Adjusted Plus/Minus (APM) [4] and Regularized Adjusted Plus/Minus (RAPM) [5].

B. APM/RAPM

The APM method [4] proposed is an improved PM method that performs a multiple regression analysis of Eq. (1) for a certain stint i (a combination of 10 players, including both allies and enemies) and expresses the coefficient β_j corresponding to each player j as his ability.

$$Y_i = \beta_0 + \sum_{j=1}^{K} \beta_j X_{ij} + \epsilon_i \tag{1}$$

The value of X_{ij} is 1 if a player of the own team is on the field, −1 if a player of the enemy team is not on the field, and 0 if a player is not on the field; Y_i is the PM per possession of the own team, and ϵ_i is the error term. However, because APM is based on simple multiple regression, it is susceptible to overlearning; therefore, RAPM is proposed [5], which is based on ridge regression.

By design, these methods are used to estimate the performance of individual players, and the results are easy to interpret. However, they have difficulty estimating the performance of lineups or player combinations, even if the players have interacted previously. Additionally, if these approaches are applied to scouting, the order of player recommendation remains the same for all teams since the information about the team does not affect the prediction results.

C. Extended RAPM

Extended RAPM [6] is a regression method that adds a lineup as an explanatory variable Z_{im} to RAPM, which performs regression only on the players, to measure lineup ability expressions and perform regression based on Eq. (2).

$$Y_i = \beta_0 + \sum_{j=1}^{K} \beta_j X_{ij} + \sum_{m=1}^{L} \gamma_m Z_{im} + \epsilon_i \tag{2}$$

In the original paper, Eq. (2) is expressed as a matrix equation. Z_{im} is a variable with a value of 1 for its own lineup, −1 for the enemy lineup, and 0 otherwise. The explained variable Y_i is not the PM score but a performance score designed to consider plays other than those related directly to the score (rebounds, etc.), the details of which are shown in Table I.

However, as explained above, this method has difficulty to apply to completely unknown lineups because lineups themselves are added to the explanatory variables. Therefore, this is not suitable as a comparison method for this study.

D. Factorization Machines

Factorization Machines [7] are based on the factorization model, which allows for the extraction of interactions between all elements and the use of those interactions as parameters. As a result, FMs can be used for sparse datasets, making them useful in many applications, particularly in recommendation systems.

Compared to polynomial models, which require N(N–1)/2 parameters when considering the interaction of any two elements from a set of N elements, FMs can reduce the number of parameters to N at most. This reduction in parameters is expected to improve the generalization performance of the model.

Furthermore, FMs can indirectly learn interactions for combinations that do not exist in the training dataset, which is not possible with other models. This capability allows FMs to perform learning on extremely sparse datasets [7].

E. DeepFM/xDeepFM

Many DNN models with FM have been proposed, among which DeepFM [11] and its advanced version, xDeepFM, are the most promising models. DeepFM successfully improves accuracy using both low-order explicit interactions based on FM and high-order implicit interaction obtained by embedding data in the DNN. In addition, xDeepFM [10] uses a compressed interaction network (CIN), which is an architecture that explicitly learns higher-order features on a vector-by-vector basis in the FM architecture; this allows more explicit interactions to be used in learning.

At the time this manuscript was written, xDeepFM is a general-purpose architecture that best matches our requirements, in our opinion; therefore, we constructed our model using this method. However, other similar methods can be used since the focus of this study is to demonstrate the effectiveness of integrating player interactions and individual player statistics.

III. Data

In this study, we collected play-by-play (PbP) and boxscore data for every game in the EuroLeague [12] and EuroCup [13] from the 2016-2021 seasons. The raw data includes information on the number of teams, players, and games, which is presented in Table II under the "Raw Data" column. To create our experimental dataset, we added three unique identifiers: game_id, team_id, and player_id. The game_id is a string that includes five components separated by underscores (_): the names of the home and away teams, the final score of the home team, the final score of the away team, and the season in which the game was played. The team_id and player_id were randomly assigned to ensure uniqueness. It should be noted that the same player_id is used for a player even if they were a member of more than one team during the data period. The PbP and boxscore data are presented in Tables III and IV, respectively.

However, Tables III and IV contain incorrect entries or do not provide some entries, which does not facilitate experimental preparation. Therefore, records in which game_id correspond to any of the following were excluded:

- Records whose game_id exists only in either boxscore or PbP.
- In boxscore, the total number of records with stater = true is not 10.
- The number of records with play_type = IN does not match the number of records with play_type = OUT.
- Records whose player_id is lost in the play_type = IN or OUT.
- Play _type = OUT records that contain a player_id that is not currently in the field.
- Play _type = IN records that contain a player_id that is already in the field.

Next, the processing applied to each dataset is described. First, the number of possessions (Poss) for a stint is calculated using Eq. (3) [14].

$$Poss = FTA \times 0.44 + 2PA + 3PA + TO \qquad (3)$$

FTA, 2PA, and 3PA indicate the numbers of free throws, 2-pt shots, and 3-pt shots attempted, respectively; and TO indicates the number of turnovers.

Additionally, we computed the performance score (PScore) for each lineup based on Table I and the extended RAPM. Subsequently, we calculated the expected performance per possession (EPP) for each lineup using Eq. (4).

$$EPP = PScore/Poss \qquad (4)$$

Here, the variance of the EEP becomes extremely high when Poss is low, which is expected because the EPP is calculated per possession. As this will result in excessive noise in the training phase, we excluded the 50 Poss threshold from the dataset. The number of players and games after the above process is shown in Table II "Processed Data" column.

TABLE I: Performance Score of Each Event

Value	Events
−1	Missed free throw, turnover, or defensive foul
−0.5	Missed shot (two or three point shots)
0.5	Assist
1	Steal, offensive or defensive rebound, block, scored free-throw, or received foul
2	Scored shot
3	Scored three-pointer

TABLE II: Number of Teams, Players, and Games

	Raw Data	Processed Data
Teams	65	
Players	1,772	938
Games	2,757	2,639

TABLE III: EXAMPLE OF PLAY-BY-PLAY DATA

Column Name	Meaning	Value example
game_id	ID of games	ALBA Berlin_Arka Gdynia_82_68_2018-19
play_number	serial number of plays	12
quarter	Quarter number (1–4)	1
team_id	ID of each team	19
marker_time	Time (10:00–0:00)	09:41
player_id	ID of player	1077
play_type	Play type	3FGM, IN, OUT, etc.
play_info	Details of the play	Three pointer (1/1–3 pt)
dorsal	Dorsal number of the player	3

TABLE IV: EXAMPLE OF BOXSCORE DATA

Column Name	Meaning	Value example
game_id	ID of games	ALBA Berlin_Arka Gdynia_82_68_2018-19
player_id	ID of player	1077
team_id	ID of each team	19
dorsal	Dorsal number of the player	3
first_name	First name	JOSH
last_name	Last name	BOSTIC
starter	Starter member or not	true
min	Time length on field	25:29
pts	Scored points	18
f2g	Success rate of 2-pt shots	2/5
f3g	Success rate of 2-pt shots	4/8
ft	Success rate of free throws	2/2
rebounds_o	Number of offensive rebounds	1
rebounds_d	Number of defensive rebounds	3
rebounds_t	Number of all rebounds	4
ast	Number of assists	2
stl	Number of steals	2
to	Number of turnovers	2
blocks_f	Number of blocks	0
blocks_a	Number of blocked	1
fouls_c	Number of commit fouls	3
fouls_d	Number of drawn fouls	2
pir	Match contribution rate	15
PM	Plus/Minus	-8

Next, based on the boxscore, we derived the annual statistics for each player in the EuroCup and EuroLeague. The statistics used, which are standard statistics used in basketball, are shown in Table Ⅴ.

TABLE V: DETAILS OF USED STATISTICS

Name	Meaning
2pt_ratio	Annual success rate of 2-pt shots
3pt_ratio	Annual success rate of 3-pt shots
ft_ratio	Annual success rate of free throws
rebounds_o	Offensive rebounds per minute
rebounds_d	Defensive rebounds per minute
rebounds_t	All rebounds per minute
ast	Assists per minute
stl	Steals per minute
to	Turnovers per minute
blocks_f	Blocks per minute
blocks_a	Blocked per minute
fouls_c	Commit fouls per minute
fouls_d	Drawn fouls per minute

IV. METHOD

Fig. 2 presents an overview of the proposed framework. It is broadly categorized into a training phase for training models and a recommendation phase for performance estimation using trained models. Although the nature of the input data differs between the training and recommendation phases, the preprocessing is the same; hence, the processor is known as a common preprocessor.

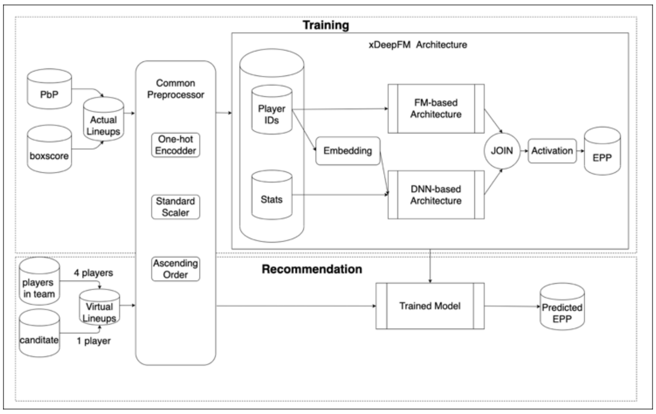

Fig.2. Overview of our proposed framework.

A. Common Preprocessor

First, a One-hot Encoder is applied to each player (IDs) such that the input format is compatible with the FM. For example, when the lineup of player groups 1–10 comprises, 1, 4, 7, 9, and 10, the data format will be sparse, as shown in Fig. 3.

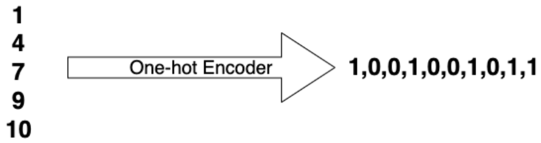

Fig. 3. Changes to sparse datasets by One-hot Encoder

Next, we apply normalization to the statistics described in Table V and to the EPP as the explained variable. To eliminate the positional dependence of the input variables, we sorted each stat from the five individuals in the ascending order. The structure of the data input to the model is shown in Fig. 4.

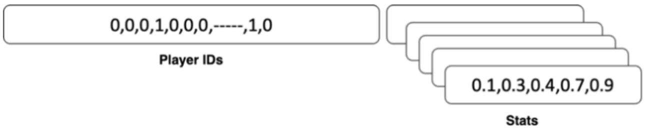

Fig. 4. Image of input data

B. Training Phase

The preprocessed dataset, consisting of actual lineups, was used to train the xDeepFM model which incorporates two primary architectures, namely FM and DNN. By applying factorization on the sparse dataset using the FM architecture, xDeepFM is expected to effectively extract player interactions. The DNN, on the other hand, was trained using the embedded representation of the sparse dataset (player IDs) and the dense dataset (player statistics). The final synthesis is expected to result in a model that exploits the advantages of both sparse and dense inputs, i.e., integrating individual player performance and their interactions.

C. Recommendation Phase

During this phase, the team's compatibility with its out-of-team players was estimated by predicting the performance of a virtually generated unknown lineup. This was done by generating virtual lineups using four players from the team's roster and one player from the rosters of other teams. These virtual lineups were then preprocessed using a common preprocessor, and the performance of the lineup was predicted using the model trained in the training phase. The predicted scores were then used to rank the recommendations, with the idea that if a lineup performed well when paired with any player from the team, then those players are necessary for the team.

V. EXPERIMENTS

A. Known Lineup Prediction

1) Experimental Design

For the methods described in the previous section, we conducted an experiment to demonstrate their high accuracy in predicting known lineups. We performed a five-fold cross-validation on the EuroCup and EuroLeague seasons from 2016 to 2019 (the first four years of the dataset) to evaluate the accuracy of the method. We compared four models: ridge regression (ridge), which is the basis of RAPM; a nonlinear kernel support vector machine (SVM) as a nonlinear model; and models trained by xDeepFM using either the player IDs (sparse dataset) only or stats (dense dataset) only (named Sparse Only and Dense Only). Notably, the Dense Only model is similar to the DNN model as it does not have an input for the FM architecture. The Root Mean Square Error (RMSE) was used as the evaluation index, and the mean and standard deviation of each fold were reported. Ridge is different from RAPM as it performs learning on lineups instead of on stints, and stats are included in the explanatory variables but are employed as alternative methods to perform comparisons under the same conditions. In the training phase, the training error is weighted by its Poss for each lineup in both models, based on the concept of extended RAPM. The hyperparameter settings for the proposed and comparative models are presented in Table VI.

TABLE VI: HYPERPARAMETERS FOR EACH MODEL

Model	Parameter
Proposed model	CIN layer size: (256, 128, 64) DNN hidden units: (128, 128) Embedding dimension: 4
Ridge	Regularization parameter: $\alpha = 1$
SVM	Regularization parameter: C$= 1$ Kernel: Gaussian kernel

2) Result

The mean accuracy and standard deviation of each fold are listed in Table Ⅶ. Our proposed model showed superior accuracy compared to the other models, followed by the Dense Only and Sparse Only models. Although Dense Only shows that the accuracy was typically based on the individual player's ability alone, the accuracy improved when the interaction between players was explicitly incorporated. Sparse Only indicated higher accuracy than the existing method, suggesting that each player's performance is not independent, and the influence of each player is important for performance estimation. As the SVM results show, capturing interactions between players is challenging for simple non-linear models.

TABLE VII: RESULTS OF LINEUP PREDICTION

Model	Mean	Std.
Proposed Model	0.202	0.0271
Ridge	0.372	0.0086
SVM	0.284	0.0104
Sparse Only	0.238	0.0107
Dense Only	0.219	0.0237
Unknown Lineup	0.215	

B. Unknown Lineup Prediction

1) Experimental Design

To validate the accuracy of our recommendations, we evaluated the predictions using an unknown lineup from the 2020-2021 season as a virtual lineup. We used each model obtained from the cross-validation for the predictions and took the average of the predictions as the final result. However, to ensure consistency with the training phase, we used only the list of players who had played in the previous four years. Therefore, lineups with less than five players due to the absence of players in the 2020–2021 season were excluded from the predictions. As a result, we predicted a total of 359 unknown lineups.

2) Result

"Unknown Lineup" in Table VI shows the estimation results. The RMSE of 0.215 implies slightly less accurate predictions for the unknown lineups compared to the known lineups, which had an RMSE of 0.202, although the difference is not significant. This factor is crucial to determine the validity of player recommendations, which is the focus of this study. However, in terms of the practicality of the recommendations, a more comprehensive discussion is necessary to determine whether they are suitable for the team's characteristics. In the following section, we provide details of our recommendation experiment.

C. Player Recommendation

1) Experimental Design

We conducted a player recommendation experiment using the learned model. First, virtual candidate lineups were generated based on the recommendation shown in Fig. 2. Based on four arbitrary players from each team's roster season and one player from the candidate group (in this case, EuroCup and EuroLeague players other than those belonging to the team) for the 2021 season, 1,715,688 lineups were created. The model predicted the team performance for the generated candidate lineups and ranked the candidate players for each team. If there are N players in a team, each candidate player was given ${}_N C_4$ results, the best of which was adopted as the candidate's performance. Similar to the unknown lineup estimation, the list of players was based on that of the previous four years, and the number of teams was limited to 32. If a player plays for more than one team in a specified year, then the total stats among all teams were used.

2) Result

The randomness of the recommendation results obtained by each team was evaluated by calculating the Spearman's rank correlation coefficients between the recommendation results of every two teams, which describe the randomness of the entire order. This was shown in Fig. 5. To calculate the rank correlation, players belonging to one of the teams were excluded from the recommendation results.

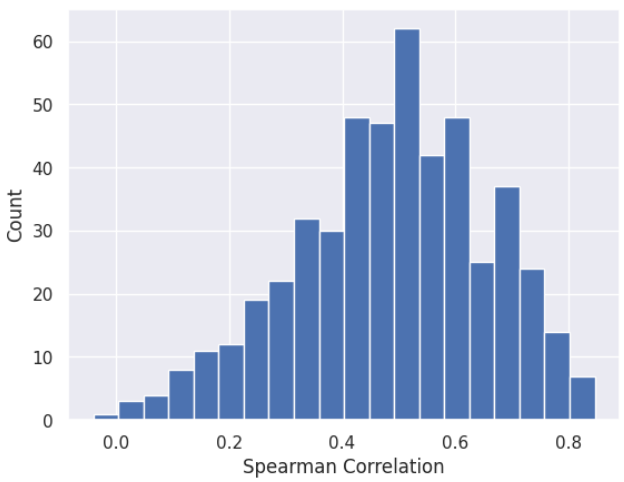

Fig. 5. Spearman's rank correlations of every two teams' recommendation results

In general, we observed a low correlation value of approximately 0.5, suggesting that only some of the teams were composed of players expected to perform particularly well, which is crucial in the recommendation process. On the other hand, a high correlation value of approximately 0.8 was observed between certain teams. This may have occurred because, in those cases, the similar characteristics of the players in the two teams led to the players who complemented those teams being similar as well.

Next, we compared the top results instead of the entire recommendations. This is because not all of the ranks are meaningful, and the results of the top ranks are of most interest when applied to actual scouting. In addition, the top results may be very similar, even if they are somewhat scattered overall. For example, if the top 20 results between two teams are perfectly matched, but the rankings of the other 267 players are entirely random, the rank correlation coefficient will be very low, although the overlap between the top 20 players is naturally 20. It is also important to check for such distortion between the results of only the top-ranked players and the results as a whole.

For this purpose, we plotted the relationship between the rank correlation coefficients among the recommended results and the number of overlaps among the top 20 players' results in Fig. 6. The correlation coefficient between the two is 0.535 (p-value << 0.01), indicating a weak positive correlation, suggesting that there is a certain relationship between the variability of the overall results among the teams and the variability of the results when focusing on the top results. This implies that the proposed framework has the important property of avoiding competition among teams for scouting targets, which is significant in practice.

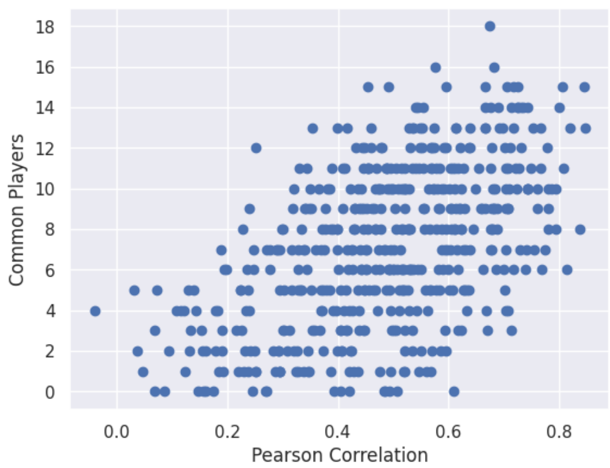

Fig. 6. The relationship between rank correlations and overlap of the top 20 players

VI. Conclusion

Overall, the proposed framework shows promising results in estimating the performance of known lineups and recommending players for unknown lineups. The integration of sparse lineup data and dense player statistics through an FM-based model appears to be effective in improving performance estimation accuracy. Furthermore, the observed variation in recommendation results suggests the potential for identifying players who may complement a team's existing players well, thereby avoiding competition with other teams for scouting targets.

Future improvements could include considering performance-level differences among leagues and teams and incorporating uncertainties such as player growth and deterioration. If widely adopted, the proposed system has the potential to make scouting more efficient and streamlined, increase players' market mobility, and reduce mismatches between teams and players, ultimately improving the competition level and revitalizing the basketball industry.

Conflict of Interest

The authors declare no conflict of interest

Author Contributions

This study was conducted as part of Haruhiko Tashiro's bachelor graduation research, supported by Nishiguchi Mao, and supervised by Fujio Toriumi.

Takuya Shimano, Kenta Morikawa, and Ryota Nagano, who are members of SportMeme, inc. advised him as sports industry specialists.

References

[1] S. Szymanski and R. Smith, "The English football industry: profit, performance and industrial structure," *International Review of Applied Economics*, vol. 11, no. 1, pp. 135-153, 1997.

[2] A. Barajas, C. M. Fernández-Jardón, and L. Crolley, "Does sports performance influence revenues and economic results in Spanish football?" *Available at SSRN 986365*, 2005.

[3] L. Hvattum, "A comprehensive review of plus-minus ratings for evaluating individual players in team sports," *International Journal of Computer Science in Sport*, vol. 18, no. 1, pp. 1-23, 2019.

[4] D. T. Rosenbaum, Measuring how NBA players help their teams win. *82games*, 2004.

[5] J. Sill, "Improved NBA adjusted+/-using regularization and out-of-sample testing," in *Proc. the 2010 MIT Sloan Sports Analytics Conference*, 2010.

[6] L. Grassetti, R. Bellio, L. D. Gaspero, G. Fonseca, and P. Vidoni, "An extended regularized adjusted plus-minus analysis for lineup management in basketball using play-by-play data," *IMA Journal of Management Mathematics*, vol. 32, no. 4, pp. 385-409, Vidoni.

[7] S. Rendle. "Factorization machines," in *Proc. 2010 IEEE International Conference on Data Mining*, 2010, pp. 995-1000.

[8] L. Deng and D. Yu, "Deep learning: methods and applications," *Foundations and Trends® in Signal Processing*, vol. 7, no. 3–4, pp. 197-387, 2014.

[9] W. Zhang, J. Qin, W. Guo, R. Tang, and X. He, Deep learning for click-through rate estimation. *arXiv preprint arXiv:2104.10584*, 2021.

[10] J. Lian, X. Zhou, F. Zhang, Z. Chen, X. Xie, G. Sun, "xDeepFM: Combining explicit and implicit feature interactions for recommender systems," in *Proc. the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018, July, pp. 1754-1763.

[11] H. Guo, R. Tang, Y. Ye, Z. Li, X. He, DeepFM: a factorization-machine based neural network for CTR prediction. *arXiv preprint arXiv:1703.04247*, 2017

[12] EuroCup. [Online]. Available: https://www.euroleaguebasketball.net/eurocup/

[13] EuroLeague. [Online]. Available: https://www.euroleaguebasketball.net/eurocup/

[14] J. Kubatko, D. Oliver, K. Pelton, and D. T. Rosenbaum, "A starting point for analyzing basketball statistics," *Journal of Quantitative Analysis in Sports*, vol. 3, no. 3, 2007.