

YuShan2023 Team Description Paper for RoboCup2023

Zekai Cheng, Yahui Ren, Chao Liu, Junmin Huang, Jingya Wang, Ye Zhu, Liang Zhang

Department of Computer Science,
Anhui University Of Technology, Ma'anshan, P.R. China
yushan2d@gmail.com

Abstract. This team description paper introduces YuShan2023's team optimization directions and methods over the past year. The passing module is one of the important modules of the team, and an effective passing strategy will improve the overall strength of the team. Since the parameters of the passing module are too complicated and the data distribution is too heterogeneous, in order to be able to classify the passing data effectively, we propose a method based on kernel density negative example learning, which improves the classification effect of different kinds of passing feature data by parsing the game log files. Based on the classification results, the key areas of the offense are extracted to realize the optimization of the decision logic of the team's passing strategy, and certain improvements are achieved in tests with several teams.

Keywords: Data Mining, Kernel Density Estimation, Negative Case Learning.

1 Introduction

Team YuShan is part of the Institute of Artificial Intelligence, School of Computer Science, Anhui University of Technology, and has participated in 8 Robot World Cup competitions since 2012, finishing third in the 2013 World Cup in Netherlands, the 2021 Online World Cup and the 2022 World Cup in Thailand. In the last 10 years, we have also won 5 championships, 4 runner-ups and 1 third place at the RoboCup China Open.

In recent years, YuShan team has reorganized its engineering and launched YuShan_Base team based on Agent-2D 3.1.0 [1] and the new version of librcsc [2]. Based on YuShan_Base for research and development, the team's characteristics are analyzed using data mining technology, and the digital twin framework is proposed and based on it, improvements are made for the team's formation and offensive and defensive state transitions as well as the players' running and passing strategies.

2 Related Work

RoboCup2D simulation soccer as a multi-intelligence collaborative AI platform [3] supports the use and testing of algorithms in popular areas of AI such as reinforcement learning, deep learning, and deep clustering, and open source projects by Helios team, Cyrus team, and Glider team in the coming years [4][5][6] have contributed greatly to the development of the 2D consortium. In recent years, with the development of artificial intelligence, many scholars have applied various artificial intelligence algorithms to the RoboCup2D simulation soccer platform.

Asali E [7] used machine learning to identify the formation of the opposing team during a game, Zare N [8] used machine learning to enhance the effectiveness of executing actions such as dribbling and passing, and Fukushima T [9] analyzed the distribution of kick actions and distinguished teams with different characteristics, indicating that the basic characteristics of a team are able to be reflected by the time and place where a certain action occurs. The basic characteristics of a team can be reflected by the time and place of a certain action. Zekai Cheng [10] built a digital twin framework, used data mining techniques to build a portrait of different teams, and used the test results to systematically analyze the results and provide a feasible direction for code advancement. Song Yuan [11] extracted and analyzed the information in RCG and RCL log files, and targeted the optimization of these feature actions based on the analysis results, and provided feedback and optimization of the code based on the optimization results.

Passing strategy is one of the important strategies of a team, which plays an important role in both team's offense and defense. In recent years, we have found in our tests that players will choose similar passing paths when attacking. Extracting the features of these passes, especially the features of some passing actions with higher gains, will greatly optimize the team's offensive logic. Section 3 describes the meth-

ods used in the experiments, Section 4 presents the test results of the experiments, and Section 5 presents the conclusions and future perspectives of the experiments. In this paper, the kernel density estimation negative example learning algorithm is used to analyze and model the passing data, and the model is used in YuShan2023.

3 Method

After collecting a large amount of test data, we found that the required discrete valid data exhibited a tendency to cluster, with larger data densities occurring within some of the course areas. For further analysis, we fitted the global distribution density of the data over the course, converting the data into a continuous distribution function is beneficial to eliminate errors and can present the data characteristics more intuitively.

Kernel density estimation is a type of non-parametric estimation, which fits the distribution of the data based on the characteristics and properties of the data itself. Kernel density parameter estimation of the distribution of passing actions aims to map the distribution of passing actions triggered on the court on a two-dimensional plane, which helps to identify the areas of the court where strategy optimization needs to be done.

A set of extracted two-dimensional point set samples $X = \{x_1, x_2, \dots, x_n\}$, the kernel density estimate at any point x_i in the plane is computed as follows.

$$f_h(x) = \frac{1}{n} \sum_{i=1}^n \frac{1}{h^d} K(x, h) = \frac{1}{nH} \sum_{i=1}^n K(x, h) \quad (1)$$

Where d is the dimensionality of the sample, h is the width of the kernel function window, and $K(\cdot)$ is a kernel function on a two-dimensional space. Here the kernel function we choose Gaussian kernel as follows.

$$K(x, h) \propto \exp\left(-\frac{x^2}{2h^2}\right) \quad (2)$$

The larger the width of the kernel function window, the larger the input area affected by the kernel function estimation calculation, and the smoother the distribution function curve, although the smooth curve is beneficial to our subsequent calculations, it

will also have an impact on the characteristics of the sample. Here the bandwidth h is automatically confirmed by Scott's Rule given by Scott D W [12], as follows.

$$h = n^{-\frac{1}{d+4}} \quad (3)$$

The kernel density estimation method converts the complicated discrete data into smooth distribution function surfaces, which avoids tedious calculations and also facilitates the calculation of subsequent feature analysis. Based on the different directions of the passes, YuShan2023 proposed a kernel density estimation negative example learning method.

We divide the sample point set into multiple classes and the point set is processed by the kernel density estimation method, which eliminates the effect of data set size differences between different point sets, while the distribution probabilities of the point set over the course have been normalized to the same magnitude. We propose the kernel density negative example learning algorithm based on the above work, YuShan2023. The algorithm uses the distribution characteristics of point sets themselves to expand the distribution probability difference between different point sets over the whole distribution area. When the probability distributions of one or two types of point sets appear significantly different from other point sets, the separated point sets are highlighted; when several types of point sets are stacked together and difficult to separate, the individual point sets are sorted hierarchically. The similar distribution between point sets is eliminated and the characteristics of point sets are strengthened, so that the probability distribution of different point sets in each region of the course forms a stratification. The specific algorithm steps are as follows.

Step 1 Divide all the passing data into n categories based on the difference of passing directions, and the data D_n of each category is then randomly divided into m subcategories to obtain $m * n$ data sets $D_{i,j}$, where $i \in (0, n)$, $j \in (0, m)$.

Step 2 Fit the distribution probability $f_{i,j}$ for each subclass.

$$f_{i,j} = \frac{1}{nH} \sum_{i=1}^n K(D_{i,j}) \quad (4)$$

Step 3 The feature extraction encoder $g(\cdot)$, $f_{i,j}$ extracts the subclass features $h_{i,j}$ by $g(\cdot)$.

Step 4. Decompose the multiclassification task into multiple binary classification tasks. The subclasses that belong to the same unified class as the learned data subclasses are data classes, and the subclasses of other classes are noise classes. The learned loss functions are as follows.

$$L_{D_{i_0,j}} = -\log \frac{\sum_{j=1}^m e^{h_{i_0,j}/\gamma}}{\sum_{i=1}^n \sum_{j=1}^m e^{h_{i,j}/\gamma_i}} \quad (5)$$

Where γ_i is the temperature coefficient of the loss function, and for the closer classes, the larger the spa coefficient is, the smoother the loss function will be. Conversely, the more distant the class, the greater the negative sample impact on the class that needs to be learned.

4 Experiments

A series of passing and carrying actions by the players of the team is considered as a chain of passing actions. The passing action is represented in the log file as a sequence of kick actions performed by different players, the first kick action is to kick the ball out by the passing player, and the second kick action is to stop the ball in his possession by applying a force opposite to the direction of the ball's speed; the carrying action is represented in the log file as a random combination of kick and dash actions repeatedly performed by a single player. The kick action accelerates the ball, the dash action is the player's own sprint, and the combination of the two actions achieves the effect of the player carrying the ball.

The match log files are divided into two types of files, RCG and RCL, in which RCG files record parameter type data, such as coordinates, speed and angle of the ball and player in each cycle, and RCL files record instruction data, such as the actions performed by the player in each cycle and the parameters of the actions. The action chain is obtained by parsing the data in the game log file, and we regard the action chain in which the action at the end of the passing action chain is a shooting action as a successful action chain, and the passing action in it is regarded as a valid passing action, as shown in Fig. 1.

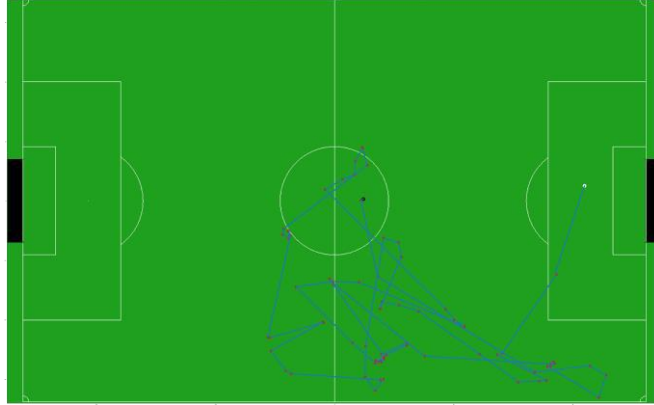


Fig. 1. Example of passing action chain

Fig. 1 shows the path of a complete passing action chain, the starting point is represented by a black dot, the ending point is represented by a white dot, and the red dot represents each passing or carrying position in the passing action chain. We consider the chain with the end action of the passing action chain as a successful action chain, and we selected HELIOS2022, CYRUS, HfutEngine2022 and MT2022 teams of 2022 World Cup Day4 as the benchmark teams to play 200 test matches. The effective passing action nodes are shown in Fig. 2.

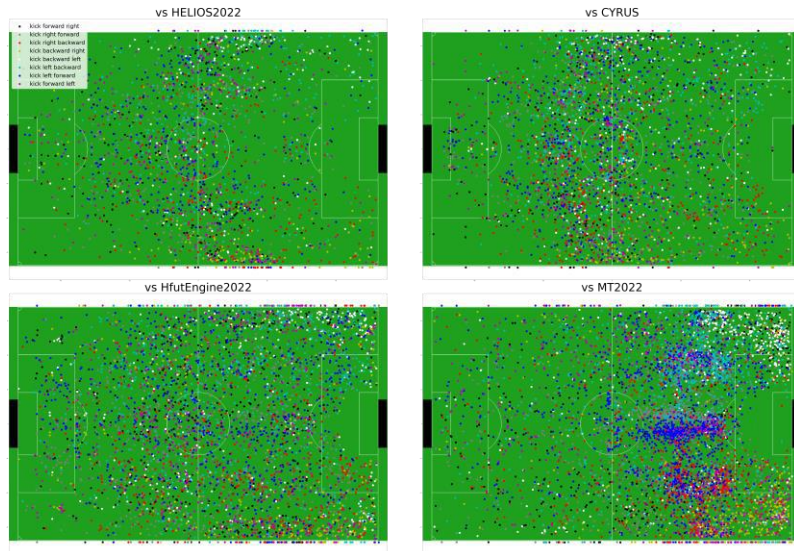


Fig. 2. Distribution of target passing action trigger locations on the court

In order to further analyze the characteristics of the passing strategy, we will focus on individual passing actions in the action chain and further analyze the characteristics of the passing actions based on the kernel density estimation negative example learning method. An example of the data feature reinforcement effect achieved by the algorithm is shown in Fig. 3.

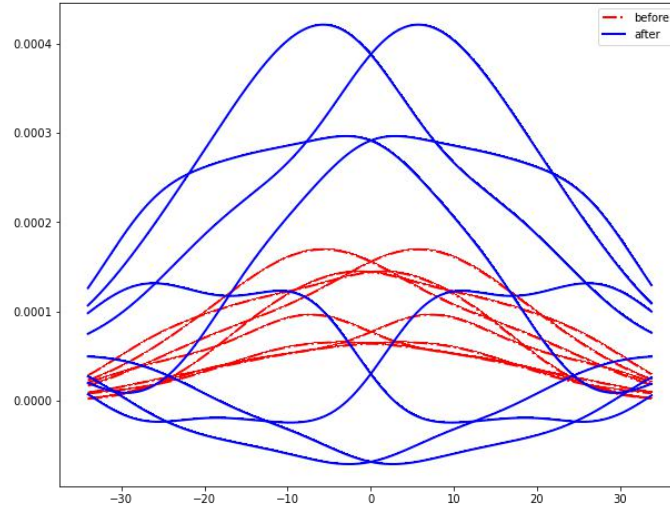


Fig. 3. The distribution function curve at $x=0$ before and after processing by the algorithm

The red dotted line is the distribution of the original data before learning, and the blue solid line is the probability distribution after learning by the kernel density negative example learning method. The method successfully stratifies the denser part on both sides and forms two trends in the middle part where dichotomies occur. The class with higher probability maintains the original features while showing the distribution features with higher trends; the class with lower probability discards the features of that part and contrasts with the other class.

The valid passing action dataset of the game is extracted, and the dataset is divided into eight class sets based on the direction of the pass, and the individual sets are processed by the kernel density negative example learning method to obtain the characteristics of the eight directions of the team's passing action on the court against different teams. As shown in Fig. 4.

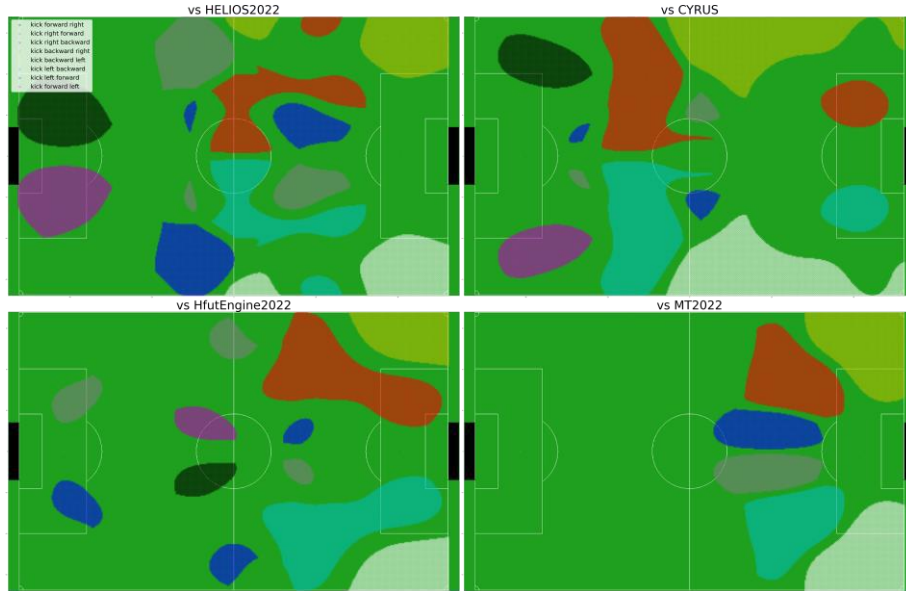


Fig. 4. Areas with high probability of different passing actions when facing different teams

Based on YuShan_base, the effective passing action data in the game are extracted as the dataset, and the characteristics of the team's passing trajectory on the court when facing different teams are obtained by the kernel density negative example learning method. As shown in Fig. 5.



Fig. 5. Path of passing preferences against different teams, with different colors indicating the data obtained from tests against different opponents

According to the experimental results, when calculating the catching points of the receiving players in the combination of passing actions, priority is given to the area with a higher probability of triggering the passing action to make the passing action more coherent between the opponent's penalty area area, for example, when the player with the ball is in the corner, the receiving player gives priority to the area on the same side of the penalty area under the condition that the passing route is unobstructed, while the free forward can move toward the central area of the penalty area This allows the player on the side of the penalty area to get the ball and immediately make the next pass to achieve the effect of continuous passing and quick attack. The choice of the position of the receiving player has a significant impact on the effect of the passing attack. For the environmental data emerging from these features, the relationship between the actions when they are triggered is expressed in the form of polynomials through methods such as particle swarm optimization algorithm, which is used as one of the judgment bases for players to judge whether to execute a certain action during the game. The collected pass data is learned for the positions of the enemy and our players, and the positions of the enemy blocking player and our player with the ball are used to calculate the best catch point in the area around the receiving player.

5 Results

The changes in the average number of goals scored, average number of goals conceded and average passing rate before and after the improvement are shown in Table 1 for 30 test matches against Team HELIOS2022, Team CYRUS, Team HfutEngine2022 and Team MT2022 of RoboCup 2022 Day4.

Table 1. Experimental results:where AvgGainChange is the change in average net goals scored before and after improvement, AvgLossChange is the change in average net goals conceded before and after improvement, and PassNumRate indicates the percentage increase in the number of desired passing actions triggered after improvement compared to that before improvement.

	avgGainBefore	AvgGainChange	avgLossBefore	AvgLossChange	PassNumRate
HELIOS2022	0.2	+0.4	7.4	-1.0	+3.89%
CYRUS	0.8	+0.6	4.6	-0.8	+16.61%
HfutEngine2022	1.8	+0.0	2.0	-0.4	+6.15%
MT2022	2.0	+1.2	1.8	-1.0	+2.94%

The improvement of the passing strategy is an optimization of the team's offensive strategy, and the effect of the improvement can be reflected by the average number of goals scored. The experiment shows that the improvement of the passing strategy has improved the average number of effective passes for each team to different degrees, which indicates the effectiveness of the improvement of the passing strategy, and also verifies the correctness of applying the negative example learning method to data analysis by kernel density estimation. In future work, if the difference in the number of samples between different categories can be balanced by the sample expansion algorithm, the effect of unbalanced samples on the analysis results can be further eliminated. Finally, we would like to thank Hidehisa Akiyama, Nader Zare, Mikhail Prokopenko and others for their open-source Base projects, which have contributed greatly to the 2D Consortium's ability to continue to advance the code, and the YuShan team for their open-source projects.

Reference

1. Akiyama, H.: agent2d-3.1.0 RoboCup tools. <https://osdn.net/projects/rctools/downloads/51943/agent2d-3.1.0.tar.gz>.
2. Akiyama, H.: Librcsc. <https://github.com/helios-base/librcsc/archive/refs/heads/master.zip>
3. Fukushima T, Nakashima T, Akiyama H. Evaluation-function modeling with multi-layered perceptron for RoboCup soccer 2D simulation[J]. Artificial Life and Robotics, 2020, 25: 440-445.
4. Zare N, Amini O, Sayareh A, et al. Cyrus2D base: Source Code Base for RoboCup 2D Soccer Simulation League[J]. arXiv preprint arXiv:2211.08585, 2022.
5. Prokopenko M, Wang P. Gliders2d: source code base for RoboCup 2D soccer simulation league[C]//RoboCup 2019: Robot World Cup XXIII 23. Springer International Publishing, 2019: 418-428.
6. Noda I, Matsubara H. Soccer server and researches on multi-agent systems[C]//Proceedings of the IROS-96 Workshop on RoboCup. 1996: 1-7.
7. Asali E, Valipour M, Zare N, et al. Using Machine Learning approaches to detect opponent formation[C]//2016 Artificial Intelligence and Robotics (IRANOPEN). IEEE, 2016: 140-144.

8. Zare N, Amini O, Sayareh A, et al. Improving Dribbling, Passing, and Marking Actions in Soccer Simulation 2D Games Using Machine Learning[C]//Robot World Cup. Springer, Cham, 2021: 340-351.
9. Nakashima T, Mifune S, Henrio J, et al. Kick extraction for reducing uncertainty in RoboCup logs[C]//International Conference on Human Interface and the Management of Information. Springer, Cham, 2015: 622-633.
10. Fukushima T, Nakashima T, Akiyama H. Similarity analysis of action trajectories based on kick distributions[C]//Robot World Cup. Springer, Cham, 2019: 58-70.
11. Cheng Zekai, Xie Ningyu, Yang Sichun, et al. Application of digital twin framework based on RoboCup simulation 2D[J]. Journal of Suzhou University of Science and Technology (Natural Science Edition), 2020.(in Chinese).
12. Song Yuan, Liu Qian, Wang Can, et al. RoboCup2D Log File Data Mining Research and Application[J]. Journal of Daqing Normal College, 2015, 35(6): 31-34.(in Chinese).