HELIOS2019: Team Description Paper

Hidehisa Akiyama[1], Tomoharu Nakashima[2], Takuya Fukushima[2], Yudai Suzuki[2], and An Ohori[2]

[1]Fukuoka University, Fukuoka, Japan
akym@fukuoka-u.ac.jp
[2]Osaka Prefecture University, Osaka, Japan
tomoharu.nakashima@kis.osakafu-u.ac.jp
takuya.fukushima@kis.osakafu-u.ac.jp
an.ohori@kis.osakafu-u.ac.jp
yudai.suzuki@kis.osakafu-u.ac.jp

Abstract. This team description paper introduces the overview of previous works and the recent research themes of the team HELIOS2019. The team is now tackling with the problem of soccer game analysis. We propose a method that adopts an approach of natural language processing and clustering in order to measure the similarity between teams. We show the results of computational experiments where the similarity between teams and players is analyzed using game log files.

1 Introduction

HELIOS2019 is a simulated soccer team for the RoboCup Soccer Simulation 2D League. The team has been participating in the RoboCup competition since 2000, and has won four championships [1].

This team description paper describes our previous efforts and current research topics. We recently focus on the opponent analysis in order to realize dynamic adaptation in the future. The remainder of this paper is organized as follows. Section 2 describes the overview of our previous works. Section 3 describes our recent approach of soccer game analysis. Section 4 provides some conclusions.

2 Previous Works

We have released a part of our team's source codes and related debugging tools in order to help new teams to participate in the competitions and to start the research of multiagent systems [2]. Currently, the released software packages are available at our project site[1].

We have proposed two important methods for developing a (simulated) robotic soccer team, a formation model using triangulation [3] and a framework of action

[1] `https://osdn.net/projects/rctools/` (Please cite [2] when you publish papers using the software packages in this site.)

sequence planning [4]. These methods have already been implemented in the released software so that it allows us to design a simulated soccer team effortlessly. Acquiring an effective evaluation function for action sequence planning is still a problem to be solved. We are trying to apply some machine learning methods for this problem [5,7].

Another successful approach of the team is an opponent formation identification [6]. In the soccer simulation 2D league, one of the essential tasks in the development of a team is to design an effective strategy. The sooner the opponent team's strategy is identified, the sooner the team can adapt its strategy in order to increase its chance to win the game. We are now tackling with the problems of soccer game analysis.

Some of our team members are working as a developer and a documentation team of the official soccer simulator. The simulator and its document are maintained on github[2] Everyone can easily contribute to them through the github site.

3 Analysis of Soccer Games using Distributed Representation of Actions and Players

We propose a method which identifies some relationship among any players in a team and can also indicate some similarities between teams. To achieve this, at the first step we use Word2vec [8,9,10] which translates actions and players into real vectors.

3.1 Proposed Method

The proposed method consists of three components: (1) to extract actions from log files recorded by the simulator, (2) to obtain the vectors of actions and players based on Word2vec, and (3) to perform clustering of actions and players in order to finally construct a distance matrix for teams.

Action extraction based on log files In the proposed method, an action is considered to be a word in a sentence, and a sequence of actions can be considered as a sentence in natural language. A series of action is defined as a sequence of actions of a single team without any intervention by the other team. Currently, we only consider pass and dribble actions and the other actions such as shoot, tackle, and intercept are ignored. A pass from the i-th player to the j-th player is defined as two subsequent kick commands by different players from the same team:

$$Pass : Kick_{(i)} - Kick_{(j)}, \tag{1}$$

A dribble of the i-th player is defined as more than one succeeding combination of a kick and a dash commands by the same player:

$$Dribble : Kick_{(i)} - Dash_{(i)} - Kick_{(i)}. \tag{2}$$

[2] `https://github.com/rcsoccersim/manual`

In the above equations, the index $i, j = 1, ..., 11$ are the player's uniform number.

Figure 1 shows an example of the extracted actions from a log file of a game. In this figure, different colors represent different players. This process enables us to obtain a corpus which includes the description of the team actions.

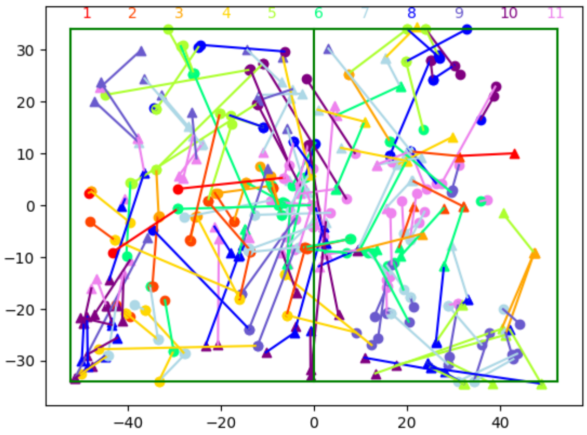

Fig. 1. Example of extracted actions.

Learning representation by Word2vec The generated corpus in the above process is used to construct a Word2vec model that converts the players to real vectors. The dimensionality of the real vector should be set large enough so that the distance between actions and players can be grasped easily by human.

Figure 2 shows an example of conversion. After processing by Word2vec, players are translated into real-valued vectors as shown in Figure 2.

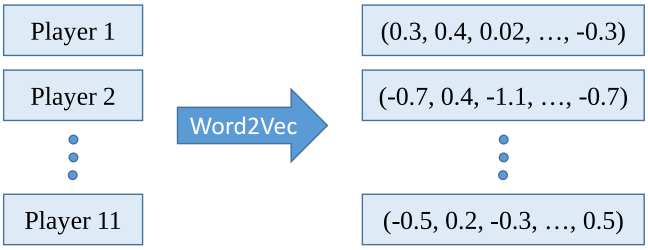

Fig. 2. Example of conversion to real vector.

Clustering For the clustering, we pay attention only to the real-valued vectors that are converted from the players in the last process. At the first step of the clustering procedure, the number of clusters k is set to 10. After the first clustering procedure, some information of players can be demonstrated. Then, according to the result of the first step, we can construct a distance matrix of teams. Finally, the second clustering result shows the similarity of teams.

Figure 3 indicates the information obtained by the clustering. We can divide players of the team into different groups by the result of first clustering, which means players in the same group have strong connections. The second clustering is performed based on the distribution of players calculated from the first clustering. Finally, the result shows which teams are similar.

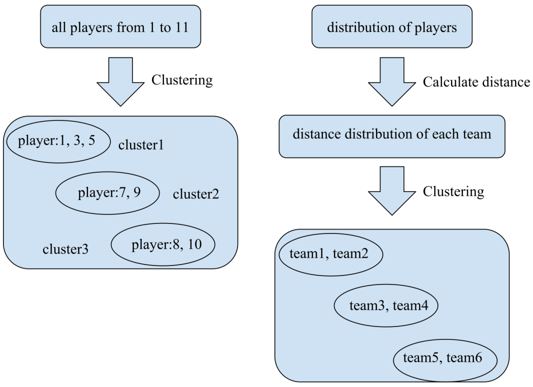

Fig. 3. Example of two processes of clustering.

3.2 Experiments

We use the log files recorded from the 2D soccer simulation matches among six teams. The 1,500 matches among the following teams are used: opuSCOM2018, agent2d, Ri-one2017, HillStone, HELIOS2017 and Fifty-Storms. Each team runs 100 rounds of games against the other teams. A corpus is generated so that it only contains the actions and players from the same team. In this experiment, we analyze each team's game separately, and then compare to each result.

The first clustering constructs a matrix that shows if any players or actions are in the same cluster (1) or not (0). The diagonal elements (i.e. the identical player or action) is set to 0. Thus, we can generate a clustering space which represents the relationship between the actions and players in the team. Table. 1 shows the clustering results for the team opuSCOM2018.

As we can see from Table 1, player 4 and player 7 are in the same cluster because plenty of passes were done between these two players in whole game logs.

Table 1. Clustering result for opuSCOM2018

Num	1	2	3	4	5	6	7	8	9	10	11
1	0	1	1	0	1	0	0	0	0	0	0
2	1	0	1	0	1	0	0	0	0	0	0
3	1	1	0	0	1	0	0	0	0	0	0
4	0	0	0	0	0	0	1	0	0	0	0
5	1	1	1	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	1	0	0	0
7	0	0	0	1	0	0	0	0	0	0	0
8	0	0	0	0	0	1	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0

In other words, it becomes clear that these two players have a strong relationship in this team. After obtaining the clustering result for all the teams, the distance of each pair of teams are calculated. Thus, a distance matrix of the teams is generated. Clustering of teams according to distance matrix reveals the similarity of teams. Tables 2 and 3 show the results of the proposed method.

Table 2. Distance between teams

	opuSCOM	Ri-one	HillStone	agent2d	HELIOS	Fifty-Storms
opuSCOM	0	5.657	4.243	1.414	3.162	5.657
Ri-one	5.657	0	3.742	4.243	2.828	4.472
HillStone	4.243	3.742	0	4.243	4.472	4.243
agent2d	1.414	4.243	4.243	0	4.243	4.0
HELIOS	3.162	2.828	4.472	4.243	0	5.831
Fifty-Storms	5.657	4.472	4.243	4.0	5.831	0

Discussion The conversion of players into real values depends on the corpus. Thus, the number of game logs and the parameters in Word2vec has an impact on the generation of the real vectors. Even if the same parameters are set, different formations of team strategy can also result in different clustering results due to the randomness in the initialization process of the neural networks in Word2vec. In our experiments, the dimension of the vectors is set to 200, and the number of training iterations is set to 100.

4 Conclusion

This paper described the previous efforts and the current research theme of HELIOS2019. The team is now mainly tackling with the problems of soccer game

Table 3. Similarity between teams

	opuSCOM	Ri-one	HillStone	agent2d	HELIOS	Fifty-Storms
opuSCOM	-	0	0	1	0	0
Ri-one	0	-	0	0	1	0
HillStone	0	0	-	0	0	1
agent2d	1	0	0	-	0	0
HELIOS	0	1	0	0	-	0
Fifty-Storms	0	0	1	0	0	-

analysis, especially opponent analysis. The proposed method uses a Word2vec, known as one of the natural language processing technique, and two step clustering. The results show the similarity between teams obtained by given game log files. We are trying to evaluate these results and to import them to the team.

References

1. Hidehisa Akiyama, Tomoharu Nakashima, “HELIOS2012: RoboCup 2012 Soccer Simulation 2D League Champion”, *RoboCup 2012: Robot Soccer World Cup XVI*, pp. 13–19, 2013
2. Hidehisa Akiyama, Tomoharu Nakashima, “HELIOS Base: An Open Source Package for the Robocup Soccer 2D Simulation”, *RoboCup 2013: Robot World Cup XVII.*, pp. 528–535, 2014.
3. Hidehisa Akiyama, Itsuki Noda, “Multi-Agent Positioning Mechanism in the Dynamic Environment”, *RoboCup 2007: Robot Soccer World Cup XI*, pp. 377–384, 2008.
4. Hidehisa Akiyama, Shigeto Aramaki, Tomoharu Nakashima, “Online Cooperative Behavior Planning using a Tree Search Method in the RoboCup Soccer Simulation”, *Proc. of 4th IEEE international Conference on Intelligent Networking and Collaborative Systems (INCoS)*, pp. 170-177, 2012.
5. Hidehisa Akiyama, Masashi Fukuyado, Toshihiro Gochou, Shigeto Aramaki, “Learning Evaluation Function for RoboCup Soccer Simulation using Humans’ Choice“, *Proceedings of SCIS & ISIS 2018*, 2018
6. Takuya Fukushima, Tomoharu Nakashima, Hidehisa Akiyama, “Online Opponent Formation Identification Based on Position Information”, *RoboCup 2017: Robot World Cup XXI.*, pp. 241–251, 2014.
7. Takuya Fukushima, Tomoharu Nakashima, Hidehisa Akiyama, “Mimicking an Expert Team through the Learning of Evaluation Functions from Action Sequences“, *RoboCup International Symposium 2018*, 2018
8. T. Mikolov, G. Corrado, K. Chen, J. Dean, “Efficient Estimation of Word Representations in Vector Space“, *Proc. of the International Conference on Learning Representations Workshop*, pp.1-12, 2013.
9. T. Mikolov, I. Sutskever, K. Chen, G. Corrado, J. Dean, “Distributed Representations of Words and Phrases and their Compositionality“, *Proc. of Advances in Neural Information Processing Systems*, pp.3111-3119, 2013.
10. Q. Le, T. Mikolov, “Distributed Representation of Sentences and Documents“, *Proc. of the International Conference on Machine Learning*, pp.1188-1196, 2014.