HELIOS2015 Team Description Paper

Hidehisa Akiyama[1], Tomoharu Nakashima[2], and Satoshi Mifune[2] and Sho Tanaka[2] and Jordan Henrio[2]

Faculty of Engineering, Fukuoka University, Japan[1]
akym@fukuoka-u.ac.jp
Department of Computer Science and Intelligent Systems, Osaka Prefecture University, Japan[2]
tomoharu.nakashima@kis.osakafu-u.ac.jp

Abstract. HELIOS2015 is a soccer simulation 2D team which has been participating in the RoboCup competition since 2000. We recently focus on the game analysis from kick records using a clustering method. This paper describes the overview of our approach for clustering game log files using Earth Mover's Distance.

1 Introduction

HELIOS 2015 is a simulated soccer team for the RoboCup soccer 2D simulation league. The team has been participating in the RoboCup competition since 2000, and has won two championships [1]. The team has released a part of their 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.

We recently focus on the game analysis using a clustering method. In this paper, we introduce our approach for clustering of game log files.

2 Related Works

We have released several open source software packages that help us to develop a simulated soccer team [2]. Now, we are mainly maintaining the following software packages[1]:

- librcsc: a base library for a simulated soccer team.
- agent2d: a sample team program using librcsc. Newbies can use agent2d as a start point for developing their own team.
- soccerwindow2: a high functional viewer, which can be used as a monitor client, a log player and a visual debugger.
- fedit2: a formation editor for agent2d. fedit2 enables us to design a team formation using human's intuitive operations.

[1] Available at: http://sourceforge.jp/projects/rctools/

They are implemented from scratch without any source codes of other simulated soccer teams. However, several idea were inspired from other released codes, such as CMUnited[3], FC Portugal[4], YowAI[5], TsinguAeolus[6], UvA Trilearn[7, 8] and Brainstormers[9, 10]. We would like to thank those teams for their effort and achievements.

In previous years, we proposed a team formation model that uses Delaunay triangulation [11] and a multiagent planning framework [12]. They have been already available in the released software.

3 Clustering of Game Log Files

We applied Earth Mover's Distance [13] for the clustering of game log files. In this section, we describe the overview of Earth Mover's Distance and its application to clustering of game log files.

3.1 Earth Mover's Distance (EMD)

Earth Mover's Distance (EMD) is a pseudo metric for measuring the distance between two probability distributions, which has been applied to various fields such as image retrieval and music information retrieval. EMD is calculated by formalizing the distance between probability distributions as a transportation problem. A transportation problem is a problem to solve the minimum transportation cost among several supplies and demands. In EMD, one probability distribution is supposed to be a supply and the other is supposed to be a demand. Then, the distance between two distributions is defined as the minimum transportation cost. A probability distribution P is represented as a weighted set of m features:

$$P = \{(p_1, w_{p1}), \ldots, (p_m, w_{pm})\}$$

where p_i is the feature vector and w_{pi} is the weight of that feature. EMD can be calculated even if the number of features of two distributions is different.

3.2 Clustering of Game Log Files using EMD

In order to calculate EMD between two game log files, a game log file has to be represented by an weighted set. We use the coordinates of the player that kicked the ball as the feature vector (p_i), and use the ball move distance by that kick as the weight (w_{pi}). Figure 1 shows the kick distribution of our test team extracted from a game against UvA Trilearn.

Hierarchical clustering is used for classifying game log files. The procedure is as follows:

1. Generate the distance matrix from EMD between game log files.
2. Calculate the distance between clusters.
3. Merge two clusters that the distance is minimum.

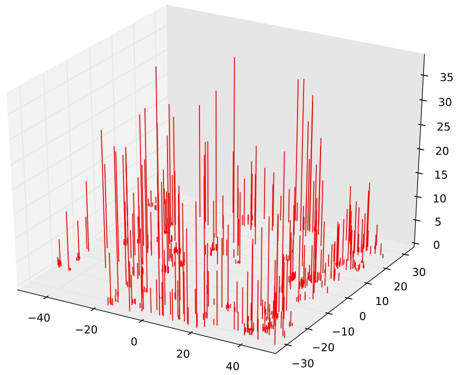

Fig. 1. Kick distribution of our test team extracted from a game against UvA Trilearn

4. Terminate the procedure if the number of clusters becomes one. Otherwise, go to 2.

Group average method is used for calculating the distance between clusters. After clustering, a dendrogram can be generated.

3.3 Experiments

We performed experiments for clustering of game log files using our method. We prepared a test team which is based on agent2d. Our test team competes against four opponent teams, UvA_Trilearn(2005), BrainStormers(2009), HELIOS(2014) and WrightEagle(2014). The number of games is 10 for each opponent team. Table 1 shows the game results.

Table 1. Game results.

Opponent Team	Win	Draw	Lose
UvA_Trilearn	10	0	0
BrainStormers	7	1	2
HELIOS	0	0	10
WrightEagle	0	0	10

Figure 2, 3 and 4 show the result of hierarchical clustering. In figures, 'U' means _Trilearn, 'B' means Brainstormers, 'H' means HELIOS, and 'W' means WrighteEagle. The characters enclosed by a red circle mean the games that our test team won. Each result is detailed in the followings.

Clustering of opponent team's kick distribution Figure 2 shows the result of hierarchical clustering for opponent teams' kick distribution. In the case of 2 clusters, UvA_Trilearn is successfully classified.

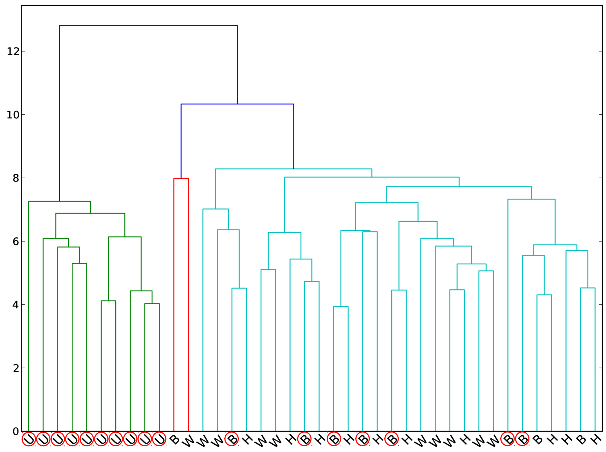

Fig. 2. Result of Clustering : opponent team's kick distribution

Clustering of our team's kick distribution Figure 3 shows the result of hierarchical clustering for our test team's kick distribution. In the case of 3 clusters, almost all defeated games belong to the central cluster and the winning games belong to other clusters.

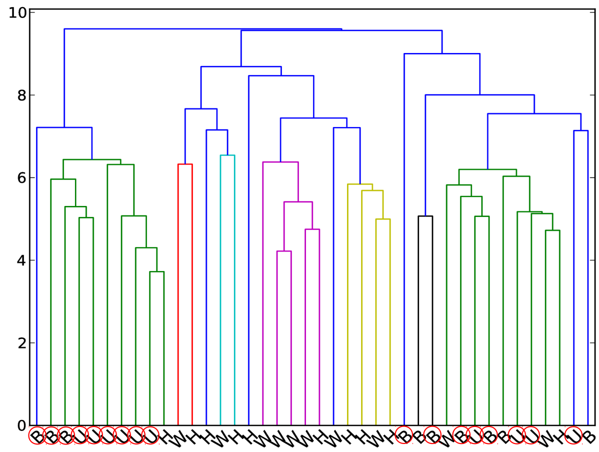

Fig. 3. Result of Clustering : our team's kick distribution

Clustering of both teams' kick distribution Figure 4 shows the result of hierarchical clustering for both teams' kick distribution. In the case of 2 clusters, all defeated games are successfully classified.

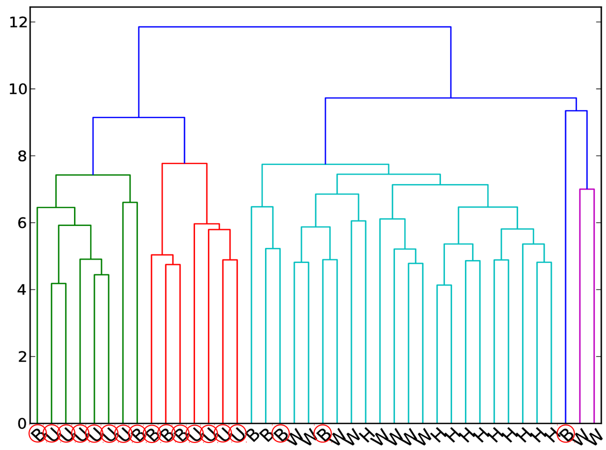

Fig. 4. Result of Clustering : Both teams' kick distribution

In the case of classification of some teams and/or tactics, it is effective to use the clustering of opponent team's kick distribution. On the other hand, in the case of classification of victory or defeat, it is effective to use the clustering of both teams' kick distribution.

4 Conclusion and Future Works

This paper described the research focus and the current effort of HELIOS2015. We proposed a method for clustering of game log files using EMD as a similarity of games. In the experiment, we could show that the specific teams and victory or defeat can be classified. The improvement of classification and online analysis are future works.

References

1. Akiyama, H., Nakashima, T.: HELIOS2012: RoboCup 2012 Soccer Simulation 2D League Champion. RoboCup-2012: Robot Soccer World Cup XVI (2013)
2. Akiyama, H., Nakashima, T.: HELIOS base: An open source package for the robocup soccer 2d simulation. RoboCup 2013: Robot World Cup XVII (2014) 528–535
3. Stone, P., Riley, P., Veloso, M.: The CMUnited-99 champion simulator team. In Veloso, M., Pagello, E., Kitano, H., eds.: RoboCup-99: Robot Soccer World Cup

III. Volume 1856 of Lecture Notes in Artificial Intelligence. Springer Verlag, Berlin (2000) 35–48

4. Reis, L.P., Lau, N., Olivéira, E.: Situation based strategic positioning for coordinating a simulated robosoccer team. Balancing Reactivity and Social Deliberation in MAS (2001) 175–197
5. Nakayama, K., Ako, T., Suzuki, T., Takeuchi, I.: Team YowAI-2002. RoboCup 2002: Robot Soccer World Cup VI (2002)
6. Yao, J., Chen, J., Cai, Y., Li, S.: Architecture of tsinghuaeolus. In: RoboCup 2001: Robot Soccer World Cup V, Springer-Verlag (2002) 491–494
7. Kok, J.R., Vlassis, N., Groen, F.: UvA Trilearn 2003 team description. In Polani, D., Browning, B., Bonarini, A., Yoshida, K., eds.: Proceedings CD RoboCup 2003, Padua, Italy, Springer-Verlag (July 2003)
8. UvA Trilearn 2003. http://www.science.uva.nl/˜jellekok/robocup/2003/
9. Riedmiller, M., Gabel, T., Knabe, J., , Strasdat, H.: Brainstormers 2d - team description 2005. In Bredenfeld, A., Jacoff, A., Noda, I., Takahashi, Y., eds.: Proceedings CD RoboCup 2005, Springer-Verlag (2005)
10. Brainstormers 2d: Brainstormers Public Source Code Release. http://sourceforge.net/projects/bsrelease/
11. Akiyama, H., Noda, I.: Multi-agent positioning mechanism in the dynamic environment. In: RoboCup 2007: Robot Soccer World Cup XI. (2008)
12. Akiyama, H., Nakashima, T., Aramaki, S.: Online cooperative behavior planning using a tree search method in the robocup soccer simulation. In: Proceedings of 4th IEEE International Conference on Intelligent Networking and Collaborative Systems (INCoS-2012). (2012)
13. Rubner, Y., Tomasi, C., Guibas, L.J.: The earth mover's distance as a metric for image retrieval. Int. J. Comput. Vision **40**(2) (November 2000) 99–121