

# Kick Extraction for Reducing Uncertainty in RoboCup Logs

Tomoharu Nakashima<sup>1</sup>(✉), Satoshi Mifune<sup>1</sup>, Jordan Henrio<sup>1</sup>,  
Oliver Obst<sup>2</sup>, Peter Wang<sup>2</sup>, and Mikhail Prokopenko<sup>3</sup>

<sup>1</sup> Osaka Prefecture University, Gakuen-Cho 1-1, Naka-Ku, Sakai, Osaka, Japan  
tomoharu.nakashima@kis.osakafu-u.ac.jp,

{satoshi.mifune, jordan.henrio}@cs.osakafu-u.ac.jp

<sup>2</sup> CSIRO Computational Informatics, Epping, NSW 1710, Australia  
{oliver.obst, peter.wang}@csiro.au

<sup>3</sup> The University of Sydney, Darlington, NSW 2006, Australia  
mikhail.prokopenko@sydney.edu.au

**Abstract.** The effectiveness of using log information in RoboCup soccer simulation 2D league is shown in this paper. Although it is not possible to exactly know a strategy that a team is taking, that strategy is well represented by how the players in the team kick during games. Extracted kicks such as passes and dribbles form a kick distribution, which hopefully represent the team's strategy. In order to show the usefulness of the kick distribution, a series of computational experiments are conducted where the uncertainty in predicting the game results is reduced by grouping the games based on the kick distributions.

**Keywords:** RoboCup · Feature extraction · Strategy analysis · Clustering · Earth Mover's Distance (EMD)

## 1 Introduction

RoboCup [1], especially RoboCup soccer simulation, is meant to be a research platform for various fields. Artificial intelligence is one of the important research fields. There have been several number of works where AI techniques were applied to RoboCup soccer simulation. For example, Riedmiller et al. [2] presented neural-network approaches for reinforcement learning that were successfully implemented in his RoboCup soccer simulation 2D team. Budden and Prokopenko [3] proposed a method that improved the precision of localization using particle filter. Nakashima et al. [4] proposed an evolutionary approach where offensive strategies evolved based on the game scores and the history of the teams' game results.

While the above mentioned works concern the learning of the soccer players, there are other aspects of RoboCup research including Naruse's work on the strategy analysis in set plays for the small robot league. This work used log files that are produced after games finish. It is intuitively possible that log information of soccer games provides useful and important insight for improving soccer teams. It can also be used to analyze the trend of the competitions over years. For example, Gabel and Riedmiller [5]

presented the analysis on the trend over the ten years of competitions as well as quantitative evaluation of team strategies. Abreu et al. [6] used log information in order to compare the robotic soccer and the human soccer, showing the similarity and dissimilarity between them. However, there are only a few works of extracting useful knowledge for improving teams from log information. In the RoboCup soccer simulation 2D league, log information is compiled in log files, which have all information on finished games such as the position of players and the ball, the actions made by players, and the communication between players plus coaches. Although there is a need for obtaining useful knowledge from those log files, effective methods have not been proposed yet. There are mainly two difficulties in dealing with log information. One is that it is unknown what kind of information to use. The other is that even if such information is successfully obtained it is still unknown how to use it.

This work proposes one solution to the first difficulty: which kind of information is useful in log files. Among the various kind of possibility, kicks are selected as the focus in this paper. Kicks in this paper include dribbles and passes only and intercepted ball kicks and clearing kicks are not considered. The purpose of this work is to show the possibility of mining log files for useful knowledge on games. A series of computational experiments are conducted to show that uncertainty in predicting the game results is reduced by the clustering of kick distributions.

## 2 Kick Distribution

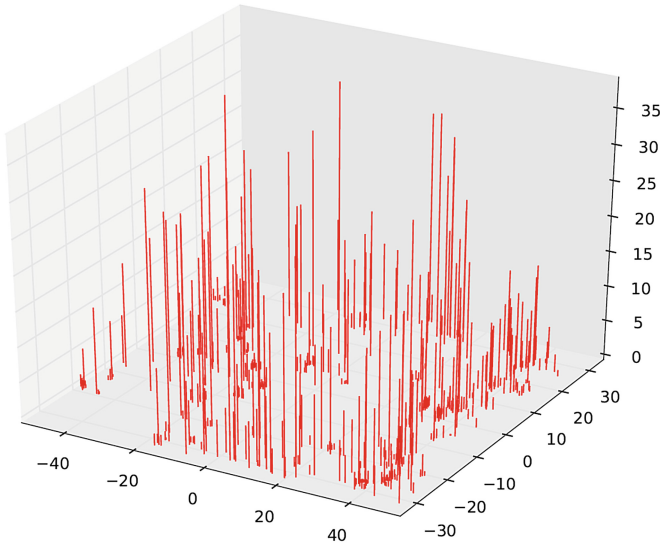
### 2.1 Log Files

A game in RoboCup soccer simulation 2D is conducted in computers. All the field information is managed by a computer process called a soccer server. It maintains the position and velocity of players and the ball, players' stamina, and it also handles the message communication between by players plus coaches. The next status of players and the ball is also calculated by the soccer server based on the actions made by the players. The information on the current status is sent from the soccer server via the UDP protocol. The actions available for players are body-turn, dash, and kick. The players must determine the direction and strength of those actions. Besides kick and dash, communication action can also be performed such as say, point, and neck-turn. All those actions are recorded as log information and dumped in log files after games are finished.

### 2.2 Kick Extraction

Dribbles and passes are extracted as kicks from log information in this paper. A dribble is defined as more than one subsequent kicks by the same player. A pass is defined as two subsequent kicks by different players from the same team. All dribbles and passes are extracted from the log files along with the distance between the two kicking points. There is no threshold in the distance for the kick extraction. Those kicks that brought the ball out of the bounds or intercepted (i.e., unsuccessful) passes by opponent teams are not extracted in this paper.

A distance between the subsequent kicks is also recorded as well as the position of the first kick. Figure 1 shows the results of kick extraction from the log file of a game between opuSCOM vs UvA\_Trilearn. In this figure, the position of the red poles indicates the place of the ball where players have kicked it, and the height shows the distance that the ball traveled until the next kick. A set of extracted kicks from one single game is called a kick distribution in this paper.



**Fig. 1.** Extracted kicks from a game between opuSCOM and UVA\_Trilearn

Each extracted kick is not separately dealt with in this paper, but whole the extracted kicks are treated as one pattern of the game. Thus, the extracted kicks form its distribution as in Fig. 1. Three types of kick distributions are generated from a single game: One is based on the extracted kicks by the right team, and another kick distribution is based on the kicks from the left team. The third kick distribution generated from a game log file is the combined distribution of the two (i.e., right and left) where all the extracted kicks regardless of the sides are brought into one single distribution.

### 3 Clustering Kick Distributions

#### 3.1 Distance Measure Between Two Kick Distributions

A distance represents a similarity between two objects. An object is assumed to be represented by a real-valued vector in most of the cases. It is generally assumed in clustering research that all vectors have the same dimensionality so that the similarity measure is calculated easily such as Euclidean distance, cosine similarity, and Manhattan distance. Our interest in this paper is in the similarity between kick distributions that are generated by extracting kicks from log files. As the number of kicks in the kick

distributions are different each other, it is highly likely that those well-known similarity measures cannot be used as it is. Instead, this paper employs Earth Movers Distance (EMD) as the similarity measure between kick distributions. EMD allows to calculate the distance between two vectors with different dimensionalities. It also allows a weight in each element of the vectors. Calculation of an EMD between two vectors with weights is done by formulating it as a transportation problem of a resource from one supplying group of cities to another demanding group of cities.

This paper calculates the distance between two kick distributions by taking an element kick with a kick distance in one kick distribution as a supplying place with available resource and also by taking an element kick in the other kick distribution as a demanding with a necessary amount of the resource.

### 3.2 Agglomerative Hierarchical Clustering

Among a number of clustering method, agglomerative hierarchical clustering algorithm is employed in this paper. Let us say there are  $N$  kick distributions  $d_1, d_2, \dots, d_N$  to be grouped by the clustering method. The following is the procedure of the agglomerative hierarchical clustering:

- Step 1. Let each kick distribution form a cluster.
- Step 2. Calculate the EMD between any pairs of clusters. The distance between two clusters is defined by the average over any pairs of the cluster elements.
- Step 3. Group the nearest two clusters and make them a new cluster.
- Step 4. Repeat Step 2. and Step 3. until all the kick distribution belong to one single cluster.

## 4 Experiments

### 4.1 Experimental Settings

In order to see if it is possible to reduce the uncertainty in predicting the results, a number of games were conducted to produce log files. Three teams are selected in the computational experiments of this paper: Gliders2014, HELIOS2014, and WrightEagle2014, which participated in the 2014 RoboCup competition.

Table 1 shows the game results after performing about ten games per opponent team. The teams that are used as opponent teams in Table 1 also participated in the RoboCup 2014 competition (Cyrus, Oxsy, UFSJ2D, and YuShan). The table also shows the uncertainty to predict the game results without knowing the opponent team name. The uncertainty is measured by the information entropy based on the number of win/lose games.

**Table 1.** Game results of the three teams

Team name	Win	Lose	Total games	Uncertainty
Gliders2014	29	24	53	0.99357
Helios2014	30	20	50	0.97095
WrightEagle2014	42	15	57	0.83147

The reason why the number of total games is different among the three teams is that some game matches did not finish properly for some technical problems that may be only solved by the developers of the teams. Thus, the total number of the games in the fourth column of Table 1. It is seen from Table 1 that Team WrightEagle has the lowest uncertainty since it won the most of the games. Gliders2014 has the highest uncertainty since the numbers of wins and losses are closest to each other.

### 4.2 Clustering Results

The uncertainty reducing process consists of the following steps: Obtaining log files, extracting kicks, and applying the hierarchical clustering. The clustering process was applied to the extracted kicks for the kick distributions obtained in Subsection 4.1. As there are three sets of kick distributions (teams of interest, their opponents, both mixed), there are nine (three times three) sets of kick distributions for the three teams. The clustering results are shown in Figs. 2, 3, 4, 5, 6, 7, 8, 9, and 10. In these figures, the forks in higher level shows that the merger of two clusters occurred in the latter process of the clustering. Thus, if three clusters are to be used from the clustering results, three higher vertical lines should be used as the corresponding three clusters.

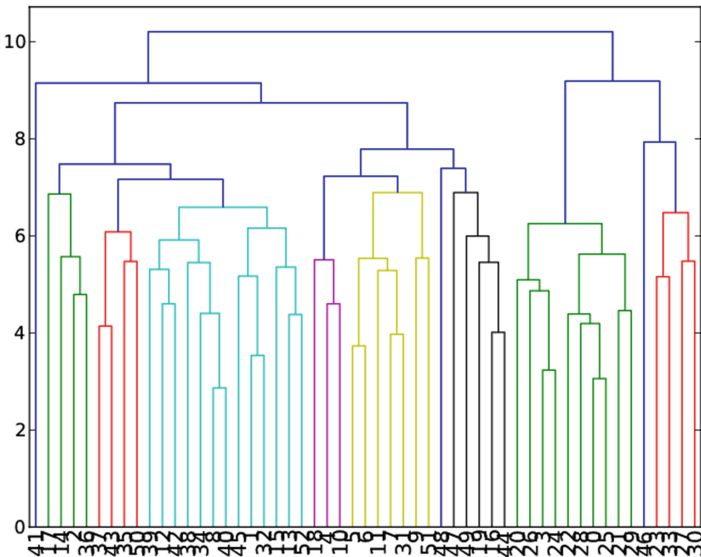


Fig. 2. Clustering results for kick distributions of Gliders2014

### 4.3 Discussions

In order to show the reduction in the uncertainty of game results, the entropy measure is calculated for each clustering results with different numbers of clusters. Tables 2, 3, 4, 5, 6, 7, 8, 9, and 10 show the uncertainty in the game results.

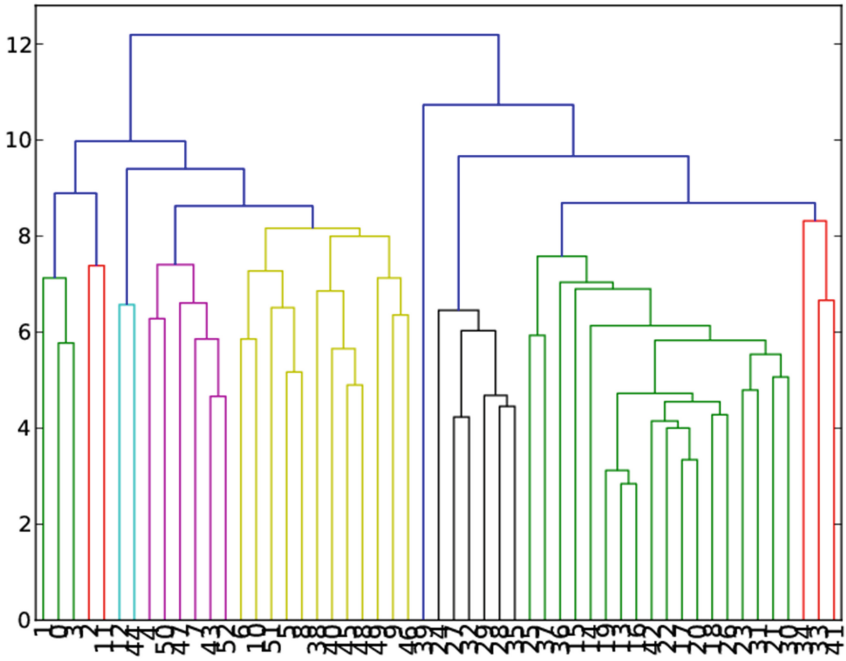


Fig. 3. Clustering results for kick distributions of Gliders2014's opponents

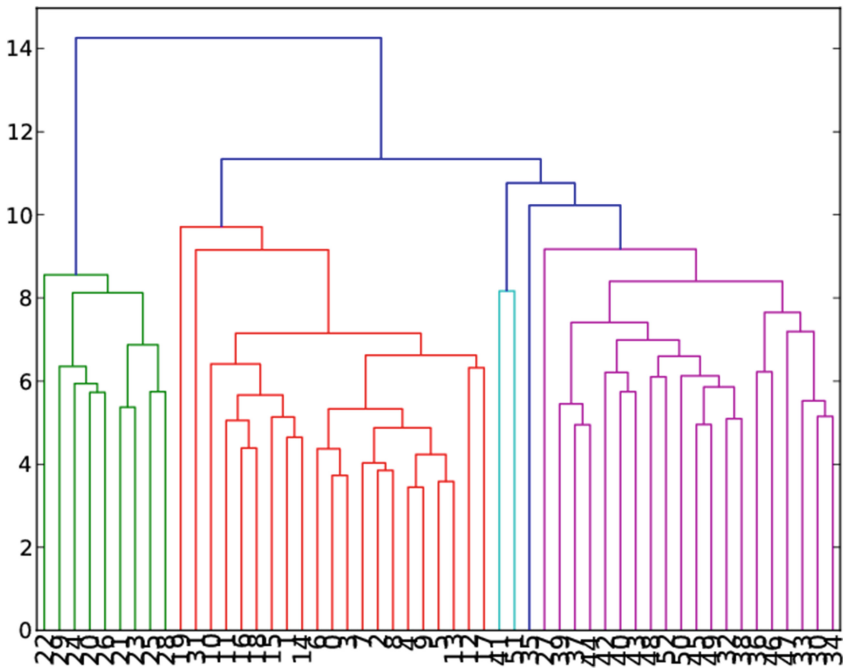


Fig. 4. Clustering results for mixed kick distributions of Gliders2014 and its opponents

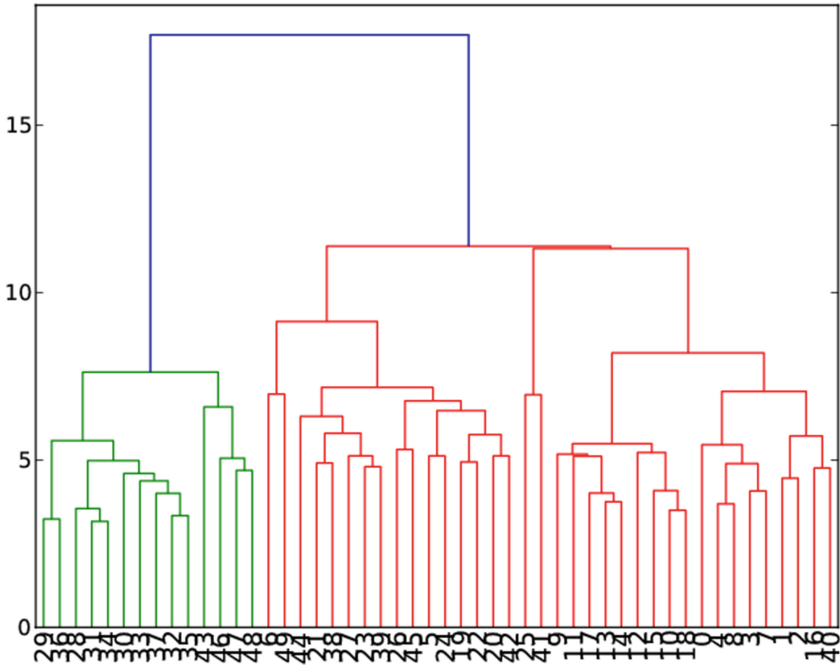


Fig. 5. Clustering results for kick distributions of HELIOS2014

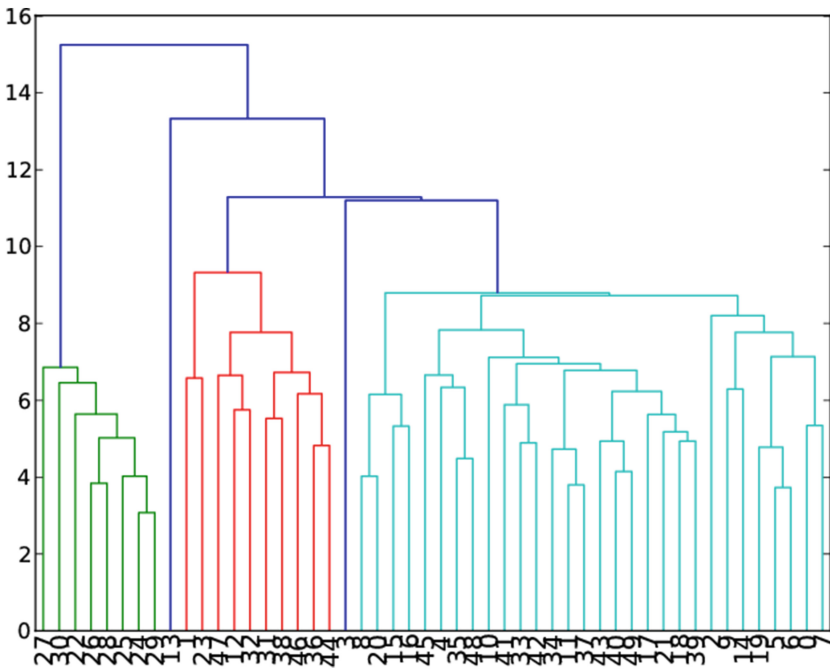


Fig. 6. Clustering results for kick distributions of HELIOS2014's opponents

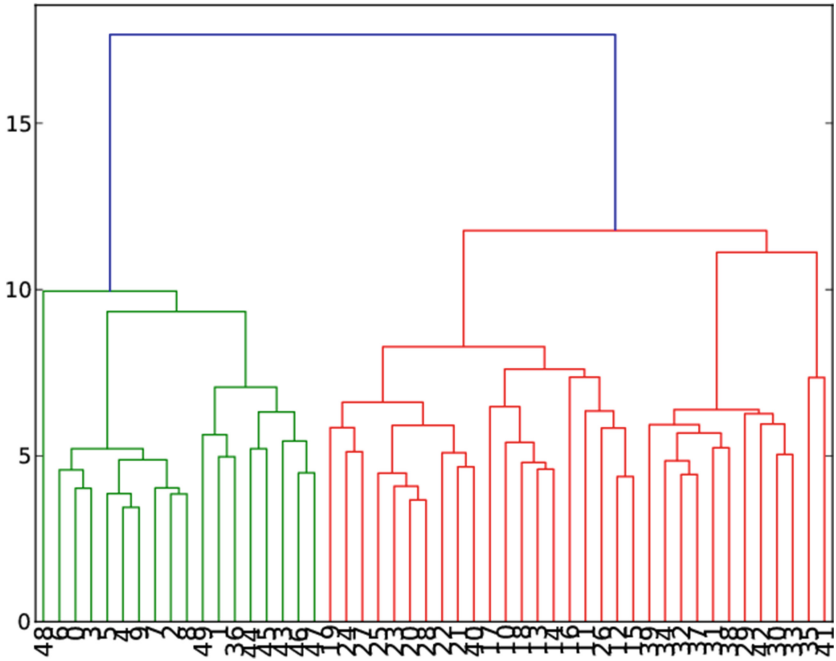


Fig. 7. Clustering results for mixed kick distributions of HELIOS2014 and its opponents

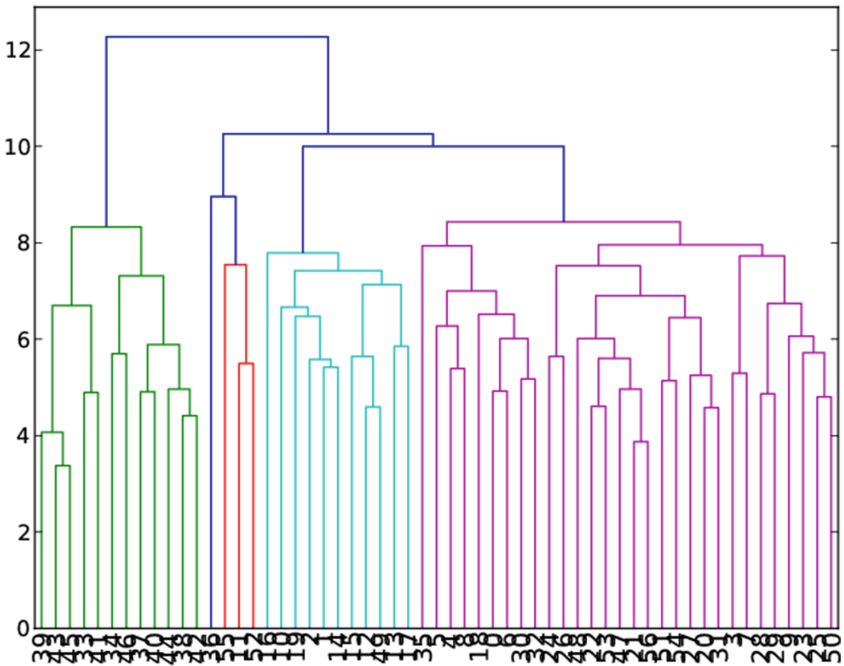


Fig. 8. Clustering results for kick distributions of WrightEagle2014



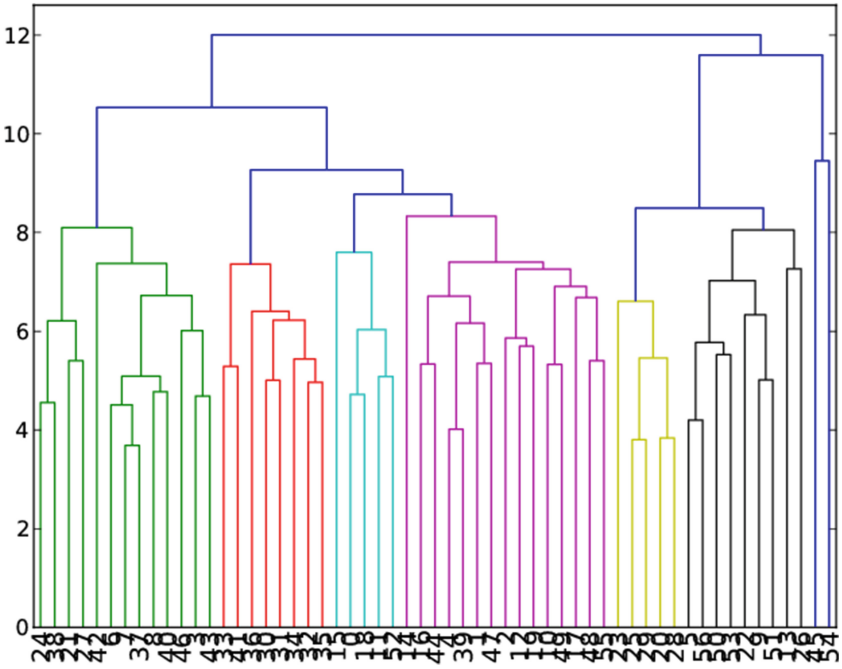


Fig. 9. Clustering results for kick distributions of WrightEagle2014's opponents

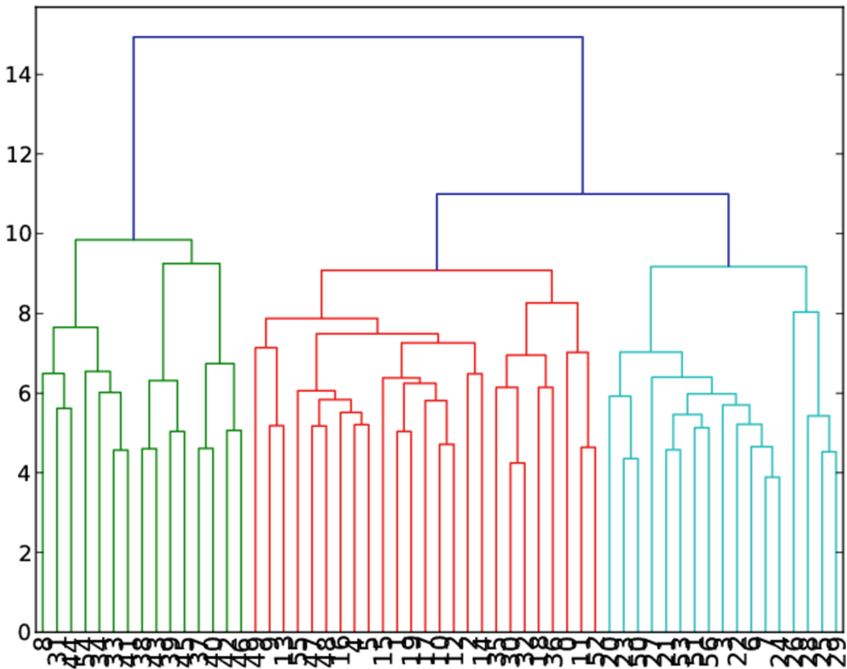


Fig. 10. Clustering results for mixed kick distributions of WrightEagle2014 and its opponents

**Table 2.** Uncertainty in game results for Gliders2014

# of clusters	Cluster 1 (Win/lose)	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Entropy
3	17 21	8 2	4 1			0.915559
4	1 0	16 21	8 2	4 1		0.893208
5	1 0	10 11	6 10	8 2	4 1	0.888028

**Table 3.** Uncertainty in game results for Gliders2014's opponent

# of clusters	Cluster 1 (Win/lose)	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Entropy
3	8 17	0 1	21 6			0.815907
4	1 4	7 13	0 1	21 6		0.809897
5	1 4	7 13	0 1	5 1	16 5	0.807928

**Table 4.** Uncertainty in game results for mixed Gliders2014 and its opponents

# of clusters	Cluster 1 (Win/lose)	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Entropy
3	9 0	8 13	12 11			0.813237
4	9 0	8 13	2 0	10 11		0.775445
5	9 0	8 13	2 0	0 1	10 10	0.757225

**Table 5.** Uncertainty in game results for HELIOS2014

# of clusters	Cluster 1 (Win/lose)	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Entropy
3	10 4	10 6	10 10			0.947093
4	10 4	10 6	2 0	8 10		0.90388
5	10 4	0 2	10 4	2 0	8 10	0.840135

**Table 6.** Uncertainty in game results for HELIOS2014's opponent

# of clusters	Cluster 1 (Win/lose)	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Entropy
3	8 0	0 1	22 19			0.81683
4	8 0	0 1	8 2	14 17		0.760191
5	8 0	0 1	8 2	0 1	14 16	0.742461

**Table 7.** Uncertainty in game results for mixed HELIOS2014 and its opponents

# of clusters	Cluster 1 (Win/lose)	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Entropy
3	6 12	15 5	9 3			0.849804
4	6 12	15 5	7 3	2 0		0.831356
5	0 1	6 11	15 5	7 3	2 0	0.819236

**Table 8.** Uncertainty in game results for WrightEagle

# of clusters	Cluster 1 (Win/lose)	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Entropy
3	11 1	2 2	29 12			0.784639
4	11 1	2 2	4 7	25 5		0.681908
5	11 1	1 0	1 2	4 7	25 5	0.660064

**Table 9.** Uncertainty in game results for WrightEagle2014’s opponent

# of clusters	Cluster 1 (Win/lose)	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Entropy
3	28 13	12 2	2 0			0.793533
4	11 2	17 11	12 2	2 0		0.761416
5	11 2	17 11	12 2	1 0	1 0	0.761416

**Table 10.** Uncertainty in game results for mixed WrightEagle2014 and its opponents

# of clusters	Cluster 1 (Win/lose)	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Entropy
3	14 1	14 11	14 3			0.727528
4	7 0	7 1	14 11	14 3		0.710828
5	7 0	3 1	4 0	14 11	14 3	0.69147

From these tables, it is seen that the uncertainty is reduced by clustering the kick distribution for any sets of kick distributions. Especially, mixing kick distributions of both teams in a game reduce the uncertainty the most for Gliders2014 and WrightEagle. This clustering results may provide useful information. For example, they can be further used for the analysis of ongoing games by the coach and for deciding whether a team changes its strategy if it is losing with a high possibility.

## 5 Conclusions

This paper presented the possible use of log files in the prediction of games. Kicks distribution is the focus of the research as a single kick itself is not enough to obtain useful information on the strategy of teams. This paper used hierarchical agglomerative clustering to generate the grouping of kick distributions. One difficulty was to measure

the similarity between two distributions due to the difference in the number of kicks. In order to overcome this issue, EMD was used. An EMD between two kick distribution is calculated by taking a kick position as the position of the place of supply/demand, and the kick distance as the quantity of the supply/demand in that place. A series of computational experiments were conducted and the results of the experiments showed that the uncertainty in the game prediction from kick distribution was reduced by the clustering results.

As this is the beginning of the research work, the next step is to further improve the reduction rate of the uncertainty. The results of this study can also be used in the actually prediction task of games. This is another future research. Another future works are to see if there are more useful features that can be extracted from log files.

## References

1. Kitano, H., Minoru, A., Kuniyoshi, Y., Noda, I., Osawa, E., Matsubara, H.: RoboCup: a challenge problem for AI. *AI Mag.* **18**(1), 73–85 (1997)
2. Riedmiller, M., Gabel, T., Hafner, R., Lange, S.: Reinforcement learning for robot soccer. *Auton. Robots* **27**(1), 55–74 (2009)
3. Budden, D., Prokopenko, M.: Improved particle filtering for Pseudo-uniform belief distributions in robot localisation. In: Behnke, S., Veloso, M., Visser, A., Xiong, R. (eds.) *RoboCup 2013. LNCS*, vol. 8371, pp. 385–395. Springer, Heidelberg (2014)
4. Nakashima, T., Ishibuchi, H., Takatani, M., Nii, M.: The effect of using match history on the evolution of RoboCup soccer team strategies. In: *The 2006 IEEE Symposium on Computational Intelligence and Games*, pp. 60–66. IEEE (2006)
5. Gabel, T., Riedmiller, M.: On progress in RoboCup: the simulation league showcase. In: Ruiz-del-Solar, J. (ed.) *RoboCup 2010. LNCS*, vol. 6556, pp. 36–47. Springer, Heidelberg (2010)
6. Abreu, P., Moreira, J., Costa, I., Castelão, D., Reis, L., Garganta, J.: Human versus virtual robotics soccer a technical analysis. *Eur. J. Sport Sci.* **12**(1), 26–35 (2011)