Cyrus 2D Simulation Team Description Paper 2018

Nader Zare [1], Mohsen Sadeghipour [2], Ashkan Keshavarzi [3], Mahtab Sarvmeili [1],
Amin Nikanjam [1], Reza Aghayari [4], Arad Firouzkoohi [4], Mohammad Abolnejad [4],
Sina Elahimanesh [4], Amin Akhgari [4]

1 Khajeh Nasir Toosi University of Technology

Nader.zare88@gmail.com, mahtab.sarvmaili@gmail.com,
nikanjam@kntu.ac.ir

2 Qazvin Azad University

mohsen.sadeghipour@icloud.com

3 Tehran University

Keshavarzi.a@ut.ac.ir

4 Atomic Energy High School

r_aghayari@yahoo.com

Abstract. Cyrus 2D Soccer Simulation was established 2012 with the aim of research and develop in multi agents systems. This year we have joined with Ziziphus for collaboration and speed up our researches. This paper express a brief description of a method for predicting player's behavior in a multi agent system using neural network with a dataset in three level (low, mid, high). The dataset was obtained from log files of past years RoboCup's matches. Behavior Prediction is used in block, mark and defensive decisions. The base code that Cyrus used is agent 3.11 [1].

Keywords: Behavior Brediction, Neural Network, Agent,
Soccer 2d simulation, Robocup

1 Introduction

Cyrus [2] robotic team was established in 2012 with the aim of research and development in fields of artificial intelligence, multi-agent decision, and deep learning. The founder of this team was the students of Shiraz University of technology but nowadays Cyrus keeps on its activities under Atomic Energy High school. Current team members are formed from the students of the K. N. Toosi University of Technology, Tehran University, Qazvin Azad University and Atomic Energy High School. Dr. Amin Nikanjam, assistant professor in the K. N. Toosi University of Technology, is the advisor

of the team. In 2018, Cyrus joined with Ziziphus [3]. Since 2013, Cyrus has taken part in Global RoboCup's Competition and in these years won 5th, 8th, 9th, 12th, and 4th places. We also took part in Iran Open Competitions since 2013 and Cyrus won first place in 2014 and challenge's first place in 2017. Cyrus won first place in Shiraz Open two times. Currently, our main focus is on deep learning algorithms, reinforcement learning algorithms and their application in 2D Soccer simulation. In this paper, we will talk about predicting kickable player's behavior.

2 Players Behavior Prediction

The main purpose of the soccer games and 2D Soccer Simulation league is to propel the ball into the opponents' goal. For this purpose, the players who get the ball should make a way to the penalty area by using pass, dribble, and finally make a shoot to the goal. Predicting this behavior have many advantages will be discussed in next section. To predict the player's behavior, we use the neural network and to train this network we generated a dataset from the RoboCup's teams and fed the neural network by this datasets. In the dataset the state of the game is categorized in three level, low-level, mid-level and high level, the label for each data is equal to the receiver of the ball. After training a neural network with high-level dataset for Helios2017 team, we can predict the target player with 90 percent probability out of two players with the high score.

2.1 Advantages of predicting the player's behavior

Predicting players' behavior is applicable in

1. Improving Team's defensive tactics by predicting opponent's behavior: The opponent behavior prediction is helpful in defense area because it could help to organize the players in defending zone.
2. Improving Team's offensive tactics by predicting opponent's behavior: Our team can mimic the decisioning of powerful teams in attack time, by using the prediction of their offensive behavior.
3. Improving Team's unmarking process by predicting teammate's behavior: our players can predict his teammate's behavior during offense situation and use this prediction to track the chain action of kickable player and the ball's path. This prediction provide successful result while players making unmark decision.

Using opponent's behavior in offense strategy already implement in Cyrus team and all the improvement will be implemented until RoboCup.

2.2 Generating offline dataset

To train the estimator neural network in three levels we need to build datasets in three levels. To generate such a dataset, we used an open source Python log extractor [4]. We generate a dataset so that every data is composed of two parts, first, the input part of the data defines the match state, second part is an 11-dimensional output, and each one

of these dimensions is assigned to a player and can be used to represent the target player in the current state. The following subsections introduce the low-level state and action space used by agents in this domain.

Low-Level, Mid-Level, High-Level State Space.

The input part in low level, mid-level and in high-level state spaces contains 92,352 and 385 features respectively. The input part in all levels is shown in table.1

Feature	Numbers of Dimension	Low-Level	Mid-Level	High-Level
Position of Ball	2	Yes	Yes	Yes
Velocity of Ball	2	Yes	Yes	Yes
Position of Players	44	Yes	Yes	Yes
Velocity of Player	44	Yes	Yes	Yes
Distance of the Ball To Important Points of the Field	9	No	Yes	Yes
Angle of the Ball To Important Point of Field	9	No	Yes	Yes
Distance of the Ball To Players	22	No	Yes	Yes
Angle of the Ball To Players	22	No	Yes	Yes
Free Angle of the Ball To Teammate	11	No	No	Yes
Distance of Players To Important Point of Field	198	No	Yes	Yes
Minimum Distance of Players to Teammate	11	No	No	Yes
Minimum Distance of Players to Opponent	11	No	No	Yes
Number of Dimension	**385**	**92**	**352**	**385**

Table 1. Feature and Dimension for input part of dataset in each level.

Fig. 1. Overview of the training data features (input part).

2.3   Training Neural Network

We have implemented rough neural network using "Object Oriented Neural Network"[5] in Cyrus2017 source code. To train the estimator neural network we also using this rough neural network. Objective oriented Neural Network is implemented in C++ and it's under open source license.

To generate the best sample of training data, 200 matches were played between Helios [6], Gliders2016 [7] and other teams from World Championships. We extracted 50000 data for training Neural Networks in three modes of low-level, mid-level and high-level in 200 games between Helios2017 and other teams. The following diagram is related to the training of the neural network in three modes that has already been discussed in section 2-2.
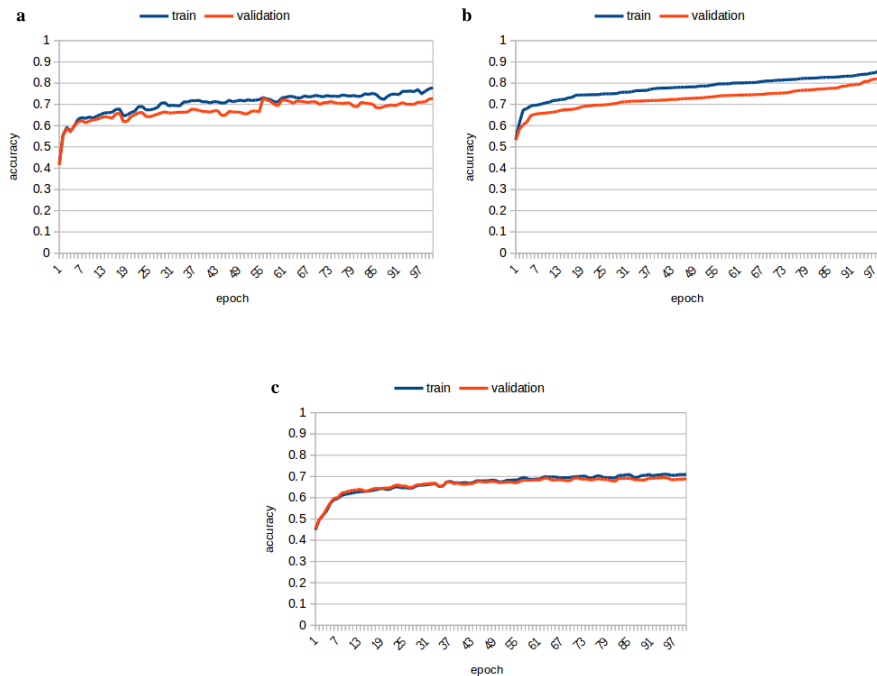

Fig. 2. Accuracy of neural network in predicting target player

The best result of the neural network given those three types of the training data for Glider2016, Helios2017 and Cyrus is shown in the following table.2. According to these results, we decide to use high-level dataset.

Team's Name	Low-Level	Mid-Level	High-Level
Gliders	78%	82%	89%
Helios	81%	84%	90%
Cyrus	74%	81%	85%

Table 2. Shows the accuracy of Behavior Prediction for Gliders, Helios, and Cyrus in each dataset's level.

2.4 Using this dataset in Cyrus team and Final Results

We use "opponent's behavior prediction" in the defensive strategy of the Cyrus. In this section, we are going to details of the algorithm and the results of the implemented neural network. In defensive strategy of our team, each one of our players will score the opponents. The scoring formula is shown in the Equation.1.

$$OpponentScore(state.unum)$$
$$= OppPosition \cdot X + \max(0.40$$
$$- Dist(OppPosition.CyrusCenterGoal)$$

$$OpponentScore(state.unum)$$
$$= OpponentScore(state.unum)$$
$$* \left(1 + NeuralNetworkOutput(unum)\right) \quad (1)$$

We calculate the score of opponents based on:

1. The closeness of the player to the goal
2. The probability of passing the ball to this player from kickable player.

After calculating this formula for each opponent, and sort them according to their score. Then we will assign mark and block behavior score to each "opponent-teammate" pair in a 2dimensional structure, then we use greedy algorithm to select our best player to mark or to block the opponent who has highest Opponent-Score, then we remove best player and marked or blocked opponent from 2dimensional structure and then we re-run the explained algorithm.

We have used the above algorithm without using "opponent behavior prediction" in Cyrus2017, in table.3 showed the result of Cyrus2017 and Cyrus2018 with other teams.

Teams	Cyrus2017	Cyrus2018
Helios2017	3.1 – 0.8	2.1 – 0.9
Glides2016	3.7 – 1.1	1.89 – 1.3
Oxsy2017	2.8 – 0.75	1.39 – 1.1

Table 3. Shows the average results of Cyrus2017 and Cyrus2018 against Helios2017, Glider2016, and Oxsy2017.

3 Future work

In our future works, we are planning to use deep reinforcement learning alongside of behavior prediction. We consider training the behavior of kickable opponent for each players and improve their deaccessioning process by using actor-critic [8] algorithm. To improve the results of the behavior prediction we want to partition the field into some geometric shapes and feed these shapes to a convolutional network to predict the behavior of players.

References

1. "Agent 2D-3.1.0 RoboCup tools - OSDN." [Online]. Available: http://en.osdn.jp/projects/rctools/downloads/51943/agent2d-3.1.0.tar.gz/. [Accessed: 22-Jan-2016].
2. Khayami Rauf, Nader Zare, Maryam Karimi, Payman Mahor, Ardavan Afshar, Mohammad Sadegh Najafi, Mahsa Asadi, Fatemeh Tekrar, Ehsan Asali, and Ashkan Keshavarzi. "CYRUS 2D simulation team description paper 2014." In *RoboCup 2014 Symposium and Competitions: Team description papers*. 2014.
3. Sadeghipour Mohsen, MohammadAli Kamkar, Azin Ghasemi, Kamyar Kalajooyeranjbar11, Khashayar Kalajooyeranjbar, Mohammad Chaposhloo, and Mohsen Firoozbakht. "RCMasterZ Team Description Paper RoboCup2017."
4. https://github.com/naderzare/LogExtractor, 2018
5. https://github.com/naderzare/OONN, 2018
6. Akiyama, Hidehisa, Tomoharu Nakashima, Katsuhiro Yamashita, and Satoshi Mifune. "Helios2013 team description paper." *RoboCup* (2013).
7. Prokopenko, Mikhail, Peter Wang, Oliver Obst, and V. Jaurgeui. "Gliders2016: Integrating multi-agent approaches to tactical diversity." In *RoboCup 2016 Symposium and Competitions: Team Description Papers, Leipzig, Germany*. 2016.
8. Hausknecht, Matthew, and Peter Stone. "Deep reinforcement learning in parameterized action space." *arXiv preprint arXiv:1511.04143* (2015).
9. Lillicrap, Timothy P., Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. "Continuous control with deep reinforcement learning." arXiv preprint arXiv:1509.02971 (2015).