Volume 2 ▪ Issue 1 ▪ March 2011

Editor-in-Chief Dr. Bekir Karlik

INTERNATIONAL JOURNAL OF ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS (IJAE)

ISSN : 2180-124X

Publication Frequency: 6 Issues / Year

CSC PUBLISHERS
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS (IJAE)

VOLUME 2, ISSUE 1, 2011

EDITED BY
DR. NABEEL TAHIR

ISSN (Online): 2180-124X

International Journal of Artificial Intelligence and Expert Systems (IJAE) is published both in traditional paper form and in Internet. This journal is published at the website http://www.cscjournals.org, maintained by Computer Science Journals (CSC Journals), Malaysia.

IJAE Journal is a part of CSC Publishers
Computer Science Journals
http://www.cscjournals.org

INTERNATIONAL JOURNAL OF ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS (IJAE)

Book: Volume 2, Issue 1, March 2011
Publishing Date: 04-04-2011
ISSN (Online): 2180-124X

IJAE Journal is a part of CSC Publishers
http://www.cscjournals.org

Published in Malaysia

Typesetting: Camera-ready by author, data conversation by CSC Publishing Services – CSC Journals, Malaysia

CSC Publishers, 2011

EDITORIAL PREFACE

The International Journal of Artificial Intelligence and Expert Systems (IJAE) is an effective medium for interchange of high quality theoretical and applied research in Artificial Intelligence and Expert Systems domain from theoretical research to application development. This is the fourth issue of volume first of IJAE. The Journal is published bi-monthly, with papers being peer reviewed to high international standards. IJAE emphasizes on efficient and effective Artificial Intelligence, and provides a central for a deeper understanding in the discipline by encouraging the quantitative comparison and performance evaluation of the emerging components of Expert Systems. IJAE comprehensively cover the system, processing and application aspects of Artificial Intelligence. Some of the important topics are AI for Service Engineering and Automated Reasoning, Evolutionary and Swarm Algorithms and Expert System Development Stages, Fuzzy Sets and logic and Knowledge-Based Systems, Problem solving Methods Self-Healing and Autonomous Systems etc.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Starting with volume 2, 2011, IJAE appears in more focused issues. Besides normal publications, IJAE intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

IJAE give an opportunity to scientists, researchers, and vendors from different disciplines of Artificial Intelligence to share the ideas, identify problems, investigate relevant issues, share common interests, explore new approaches, and initiate possible collaborative research and system development. This journal is helpful for the researchers and R&D engineers, scientists all those persons who are involve in Artificial Intelligence and Expert Systems in any shape.

Highly professional scholars give their efforts, valuable time, expertise and motivation to IJAE as Editorial board members. All submissions are evaluated by the International Editorial Board. The International Editorial Board ensures that significant developments in image processing from around the world are reflected in the IJAE publications.

IJAE editors understand that how much it is important for authors and researchers to have their work published with a minimum delay after submission of their papers. They also strongly believe that the direct communication between the editors and authors are important for the welfare, quality and wellbeing of the Journal and its readers. Therefore, all activities from paper submission to paper publication are controlled through electronic systems that include electronic submission, editorial panel and review system that ensures rapid decision with least delays in the publication processes.

To build its international reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJAE. We would like to remind you that the success of our journal depends directly on the number of quality articles submitted for review. Accordingly, we would like to request your participation by submitting quality manuscripts for review and encouraging your colleagues to submit quality manuscripts for review. One of the great benefits we can provide to our prospective authors is the mentoring nature of our review process. IJAE provides authors with high quality, helpful reviews that are shaped to assist authors in improving their manuscripts.

Editorial Board Members
International Journal of Artificial Intelligence and Expert Systems (IJAE)

EDITORIAL BOARD

TABLE OF CONTENTS

Volume 2, Issue 1, March 2011

Pages

Learning of Soccer Player Agents Using a Policy Gradient Method : Coordination Between Kicker and Receiver During Free Kicks

Harukazu Igarashi arashi50@sic.shibaura-it.ac.jp
Information Science and Engineering
Shibaura Institute of Technology
Tokyo, 135-8548, Japan

Koji Nakamura k-naka@rio.odn.ne.jp
Information Science and Engineering
Shibaura Institute of Technology
Tokyo, 135-8548, Japan

Seiji Ishihara ishihara@hiro.kindai.ac.jp
Electronic Engineering and Computer Science
Kinki University
Hiroshima, 739-2116, Japan

Abstract

As an example of multi-agent learning in soccer games of the RoboCup 2D Soccer Simulation League, we dealt with a learning problem between a kicker and a receiver when a direct free kick is awarded just outside the opponent's penalty area. We propose how to use a heuristic function to evaluate an advantageous target point for safely sending/receiving a pass and scoring. The heuristics include an interaction term between a kicker and a receiver to intensify their coordination. To calculate the interaction term, we let a kicker/receiver agent have a receiver's/kicker's action decision model to predict a receiver's/kicker's action. Parameters in the heuristic function can be learned by a kind of reinforcement learning called the policy gradient method. Our experiments show that if the two agents do not have the same type of heuristics, the interaction term based on prediction of a teammate's decision model leads to learning a master-servant relation between a kicker and a receiver, where a receiver is a master and a kicker is a servant.

Keywords: RoboCup, Soccer Simulation, Multiagents, Policy-Gradient methods, Reinforcement Learning.

1. INTRODUCTION

Recently, much work is being done on the learning of coordination in multi-agent systems [1, 2]. The RoboCup 2D Soccer Simulation League [3] is recognized as a test bed for such research because there is no need to control real robots and one can focus on learning coordinative behaviors among players. However, multi-agent learning continues to suffer from several difficult problems such as state-space explosion, concurrent learning [4], incomplete perception [5], and credit assignment [2]. In the games of the Robocup 2D Soccer Simulation League, the state-space explosion problem is the most important and difficult. Solving it is the main objective of this paper. However, the other three problems should be considered in varying degrees.

As an example of multi-agent learning in a soccer game, we dealt with a learning problem between a kicker and a receiver when a direct free kick is awarded just outside the opponent's penalty area. The kicker must make a shoot or pass the ball to the receiver to score a goal. To which point in the soccer field should the kicker kick the ball and the receiver run in such a situation? We propose how to use a heuristic function to evaluate an advantageous target point for safely sending/receiving a pass and scoring. The heuristics include an interaction term between a kicker and a receiver to intensify their coordination. To calculate the interaction term, we let a kicker/receiver agent have a receiver's/kicker's action decision model to predict a receiver's/kicker's action. The soccer field is divided into cells whose centers are candidate targets of a free kick. The target point of a free kick is selected by a kicker using Boltzmann selection with the heuristic function. The heuristic function makes it possible to handle a large space of states consisting of the positions of a kicker, a receiver, and their opponents. Parameters in the function can be learned by a kind of reinforcement learning called the policy gradient method. The point to which a receiver should run to receive the ball is concurrently learned in the same manner.

We found the following two points from our learning experiments. First, if a kicker and a receiver have the same type of heuristics in their action evaluation functions, they obtain policies similar to each other by learning and that makes both action decisions agree well. Second, if the two agents do not have the same type of heuristics, the interaction term based on prediction of a teammate's decision model leads to learning a master-servant relation between a kicker and a receiver, where a receiver is a master and a kicker is a servant. This paper will present some clues to multi-agent learning problems through solving this type of free-kick problems.

2. COORDINATION OF SOCCER AGENTS

2.1 Cooperative Play in RoboCup 2D Soccer Simulation

Reinforcement learning is widely used [6,7] in the research areas of multi-agent learning. In the RoboCup 2D Soccer Simulation League, Andou used Kimura's stochastic gradient ascent (SGA) method [8] to learn the dynamic home positions of 11 players [9]. Riedmiller et al. applied TD learning to learn such individual skills as intercepting the ball, going to a certain position, or kicking and selecting those individual skills [10]. They dealt with attacking problems with 2v1 (2 attackers and 1 defender), 2v2, 3v4, and 7v8. Stone et al. studied keepaway problems with 3v2 [11] and half-field offense problems with 4v5 [12] using Sarsa [6] to learn the selection of macro behaviors such as ball holding, passing, dribbling, and shooting.

2.2 Coordination at Free Kicks

In the previous section, we cited several researches on the cooperative behaviors of soccer agents. However, a crucial problem remains. In their research, each agent apparently learns its policy of action selection "autonomously" to complete the received task. However, Riedmiller et al. assumed that all agents share input information, which are the x-y positions of all players and the ball, with other agents [10]. Stone et al. used other agents' experiences, which are time-series data on state, action, and reward, to accelerate learning in a large problem [12]. For that purpose, agents must communicate their experiences to their partners to facilitate information sharing among themselves. If agents share input information or experiences with other agents, all agents will obtain the same value function by learning. That will simplify the realization of various cooperative plays among agents. However, if an agent's observation is imperfect or uncertain as in games of the RoboCup 2D Soccer Simulation League, all agents cannot share the same input information with other agents. If communication between agents is not perfect, they cannot share their experiences with other agents. Moreover, if only agents that have identical value functions are assumed, agent individuality and division of roles among them may not emerge from agents' learning. In the next section, we propose a method where all agents learn autonomously without assuming perfect communication or identical input information.

3. LEARNING SOCCER AGENTS BY A POLICY GRADIENT METHOD

3.1 Policy Gradient Method

A policy gradient method is a kind of reinforcement learning scheme that originated from Williams' REINFORCE algorithm [13]. The method locally increases and maximizes the expected reward per episode by calculating the derivatives of the expected reward function of the parameters included in a stochastic policy function. This method, which has a firm mathematical basis, is easily applied to many learning problems. It is extended by Kimura to learning problems in Partially Observable Markov Decision Processes (POMDPs) , which is known as Stochastic Gradient Ascent (SGA) method [8]. Moreover, Igarashi et al. proved that a policy gradient method can be applied to learning problems in non-Markov Decision Processes [14,15]. They applied it to pursuit problems where the policy function consists of state-action rules with weight coefficients that are parameters to be learned [15]. In this paper, we take this approach for agents to learn how to make their action decisions.

3.2 Stochastic Policy for Action Decision

In this section, we propose a stochastic policy for determining kicker's and receiver's actions during direct free kicks. We divide the opponent's penalty area into 32 cells (5m×5m) and assume additional three cells (5m×4.5m) inside the goal net area, as shown in Fig. 1. A_{cell} denotes the set of center points of these 35 cells. Selecting a kicker's/ receiver's action $a_K/a_R(\in A_{cell})$ is defined as selecting a cell to the center of which a kicker/receiver should kick the ball or run. If the two agents select the same cell, i.e. $a_K=a_R$, their intentions agree well with each other and a pass between them would succeed with a high possibility.

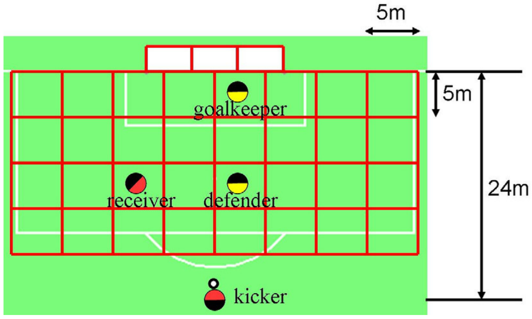


FIGURE 1: Example of Player Arrangements.

We consider objective function $E_\lambda(a_\lambda;\, s,\{\omega^\lambda_j\})$ (≤ 0) of an agent such as

$$E_\lambda\left(a_\lambda;s,\omega^\lambda\right)=-\sum_i \omega_j^\lambda \cdot U_j^\lambda\left(a_\lambda;s\right) \ , \tag{1}$$

where functions $U_j^\lambda(a_\lambda;s)$ $(\ge 0)$are the j-th heuristics that evaluate action $a_\lambda(\in A_{cell})$. Symbol λ $(\in\{K,R\})$ indicates a type of agent, where K/R means a kicker/receiver. State s $(\in S)$ means a state of the whole multi-agent system, which includes information of all players on the field and the ball. A set of parameters $\{\omega^\lambda_j\}$ is denoted simply by ω^λ in the left-hand side of (1). Next, we define agent's policy π_λ $(a_\lambda;s,\, \omega^\lambda)$ by a Boltzmann distribution function:

$$\pi_\lambda\left(a_\lambda;s,\omega^\lambda\right)\equiv\frac{e^{-E\left(a_\lambda;s,\omega^\lambda\right)/T}}{\sum_{x\in A_{cell}} e^{-E\left(x;s,\omega^\lambda\right)/T}} \ . \tag{2}$$

Note that action a_λ with a lower value of $E_\lambda(a_\lambda;\, s,\omega^\lambda)$, which means an action with a higher value of $U_j^\lambda(a_\lambda;s)$, is selected by (2) with a higher possibility. Weight parameters ω^λ in (1) and (2) are determined by a policy gradient method summarized in the next section.

3.3 Autonomous Action Decision and Learning

For the autonomous action decisions and the learning of each agent, we approximate policy function $\pi(a;s)$ for the whole multi-agent system by the product of each agent's policy function $\pi_\lambda\,(a_\lambda;s,\, \omega^\lambda)$ in (2) as [15,16]

$$\pi(a;s) \approx \prod_\lambda \pi_\lambda\left(a_\lambda;s,\omega^\lambda\right) \,, \tag{3}$$

where a=(a_K,a_R).

In (3), it seems that the correlation among agent action decisions is neglected. However, each agent can see all other agent states and use them in its policy function $\pi_\lambda\,(a_\lambda;s,\, \omega^\lambda)$. Thus, the approximation in (3) will contribute to learn coordination among agents. Note that each agent cannot get perfect information on state s and communication among agents is limited in games of the RoboCup 2D Soccer Simulation League. Therefore, agents cannot get accurate and complete information on the whole system nor share their observations with each other perfectly.

At the end of each episode, common reward r is given to all agents after evaluating results and behaviors of the whole agent system. That is a promising idea to avoid causing concurrent learning problems. The derivative of expectation of reward E[r] for parameter ω^λ is given as

$$\frac{\partial E[r]}{\partial \omega^\lambda} \approx E\left[r\sum_{t=0}^{L-1} e_{\omega^\lambda}(t) \right] \,, \tag{4}$$

if we use (3) and assume that $E_{\lambda'}(a_{\lambda'};s)$ $(\lambda' \neq \lambda)$ does not depend on ω^λ. L is the size of an episode.

With (3), characteristic eligibility $e_\omega$on the right-hand side of (4) can be written as [15,16]:

$$e_{\omega^\lambda}(t) \equiv \frac{\partial}{\partial \omega^\lambda} \ln \pi_\lambda\left(a_\lambda(t);s(t),\omega^\lambda\right), \tag{5}$$

where $a_\lambda(t)$ and $s(t)$ are the action and state of agent λ at discrete time t.

The derivative of E[r] in (4) leads to the learning rule of parameters ω^λ as

$$\Delta\omega^\lambda = \varepsilon \cdot r \sum_{t=0}^{L-1} e_{\omega^\lambda}(t) \,, \tag{6}$$

where ε is a positive small number called learning ratio. Let assume that each agent makes an action decision by policy π_λ in (2) only once at the start of every episode, i.e. at t=0, and updates ω^λ by the learning rule in (6) at the end of every episode. The learning rule for agent λ is given by

$$\Delta\omega_j^\lambda = \varepsilon \cdot r \cdot \frac{1}{T}\left[U_j^\lambda(a_\lambda;s) - \sum_{x\in A_{cell}} U_j^\lambda(x;s)\pi_\lambda\left(x;s,\left\{\omega_j^\lambda\right\}\right) \right] \cdot \tag{7}$$

4. FREE-KICK PROBLEM AND LEARNING EXPERIMENTS

4.1 Arrangement of Players

We only consider the part of a soccer field used in the RoboCup 2D Soccer Simulation League. A kicker and a receiver are agents that learn team play during free kicks. There are a defender and a goalie that are opponents, who do not learn anything. An example of the arrangement of the four players is shown in Fig. 1. The origin of the coordinate axes is located at the center mark of the soccer field. The x/y-axis is set parallel to the touch/goal line.

A kicker is only given the role of kicking the ball during direct free kicks. The x-coordinate of the free-kick position is fixed to 24 m from the opponent goal line, while the y-coordinate is selected at random. A receiver is assumed to run to receive the pass and to immediately shoot toward the goal. The receiver is set at random to a cell that is not an offside position and not any of the three cells behind the goal's mouth. An opponent goalie/defender is randomly located in the goal/penalty area. The two defenders try to intercept the pass and thwart the two offense agents. In addition to the four players, a coach-client called a "trainer" changes the play mode from kickoff to free kick, sets the players and the ball, watches the game, and informs the two agents whether a kicker's pass was safely received and whether their shots successfully scored a goal.

4.2 Heuristics Used in Objective Function

In learning experiments, four kinds of heuristic functions $\{U_i^{\lambda}(a_{\lambda})\}$ $(i=1,2,3,4)$ were used to evaluate the suitability of the selected cell for kickers and receivers. In this section, let cell k be the one selected by kicker's/receiver's action a_{λ}. $U_1^{\lambda}(a_{\lambda};s)$ considers the existence of opponents in the pass direction. $U_2^{\lambda}(a_{\lambda};s)$ expresses heuristics where shooting from nearer to the goal mouth has a greater chance of scoring a goal. $U_3^{\lambda}(a_{\lambda};s)$ evaluates a distance between the center of cell k and the nearest opponent to the center of cell k. $U_4^{\lambda}(a_{\lambda};s)$ considers the distance between the center of cell k and the current receiver's position.

U_1^{λ}, U_3^{λ} and U_4^{λ} are heuristics for passing the ball safely, while U_2^{λ} is heuristics for making an aggressive pass. These four functions are normalized to avoid exceeding 10.0 and their strict definitions are given in Appendix.

4.3 Interaction Between Two Action Decisions

In this section, we introduce an interaction between the action decisions of a kicker and a receiver. If any discrepancy exists in the information that the two agents get or perceive, learning the cooperative play described in Section 3 may be difficult. In actual human soccer games, all teammates cannot be expected to share identical information and heuristic knowledge to make their decisions. For this reason, we consider another heuristics, U_5, an interaction term. That makes one agent select its action that fits well an action selected by a teammate agent. The interaction term helps accelerate cooperation between the two agents. We define this interaction term by function $U_5(a_K,a_R;s)$:

$$U_5\left(a_K,a_R;s\right) \equiv \left(-\left|X_{KR}\right|-\left|Y_{KR}\right|+50.0\right)/5.0\,, \tag{8}$$

where a_K/a_R is kicker's/receiver's action and (X_{KR},Y_{KR}) is a difference vector between a cell to which a kicker intends to kick the ball and a cell to which a receiver runs. Adding interaction term $U_5(a_K,a_R;s)$ in (8) to the objective function in (1) , we use the following objective functions $E_K(a_K;s, \omega^K)$ and $E_R(a_R; s,\omega^R)$ for a kicker and a receiver:

$$E_K\left(a_K;s,\omega^K\right) = -\sum_i^4 \omega_i^K \cdot U_i^K\left(a_K;s\right) - \omega_5^K \cdot U_5\left(a_K,a_R^*;s\right)\,, \tag{9}$$

$$E_R\left(a_R;s,\omega^R\right) = -\sum_i^4 \omega_i^R \cdot U_i^R\left(a_R;s\right) - \omega_5^R \cdot U_5\left(a_K^*,a_R;s\right)\,. \tag{10}$$

To calculate interaction term $U_5(a_K,a_R;s)$ in (8), a kicker/receiver needs information on the action that the other teammate is going to select. One solution is sending the information by say command. But completely assuming the sending and receiving of all teammate actions is neither realistic in actual human soccer games nor desirable even in games of the RoboCup 2D Soccer Simulation League. In this paper, we adopt a method in which an agent has the other teammate's action-decision model inside itself and uses it for predicting the teammate's next action without asking it of the teammate. Thus receiver's action a^*_R in (9) is an action predicted by a kicker, and kicker's action a^*_K in (10) is an action predicted by a receiver. The next section describes how to predict a teammate's action.

4.4 Prediction of a Teammate's Action

Let us discuss a kicker's action decision. Objective function $E_K(a_K;s,\omega^K)$ in (9) is used to determine kicker action a_K. Function $E_K(a_K;s,\omega^K)$ includes receiver's action a^*_R in its interaction term $U_5(a_K,a^*_R;s)$ shown in (8). Receiver's action a^*_R should be determined by minimizing receiver objective function $E_R(a_R;s,\omega^R)$ shown in (10). Function $E_R(a_R;s,\omega^R)$ includes kicker action a^*_K in $U_5(a^*_K,a_R;s)$. That is, determining kicker's action a_K needs receiver's action a_R and vice versa. To break this endless loop, we use receiver's/kicker's action a^*_R/a^*_K predicted by minimizing $E_R(a_R;s,\omega^R)/E_K(a_K;s,\omega^K)$ that has no interaction term $U_5(a_K, a_R;s)$: i.e.,

$$a_R^* = \arg\min_{a_R}\left[-\sum_{i=1}^{4}\omega_i^R U_i^R\left(a_R;s\right)\right], \qquad (11)$$

$$a_K^* = \arg\min_{a_K}\left[-\sum_{i=1}^{4}\omega_i^K U_i^K\left(a_K;s\right)\right] \qquad (12)$$

for calculating the right-hand sides of (9) and (10). This method represents that an agent predicts another agent's action using the other agent's action-decision model. The receiver and kicker models are represented by weight coefficients $\{\omega_i^R\}$ $(i=1,2,3,4)$ and $\{\omega_i^K\}(i=1,2,3,4)$ in (11) and (12), respectively. However, the values of the coefficients are updated every episode during learning. To make the prediction as accurate as possible, a kicker and a receiver teach each other the values of their own weight coefficients every ten episodes by say command in our experiments. This helps update a teammate's action-decision model and keeps it current. This teaching in games is not necessary at all if agents are not learning their policy functions.

4.5 Reward

We deal with a cooperative play between a kicker and a receiver during direct free kicks. For learning this cooperative play, a large reward must be given to the two agents only if a kicker's pass is successfully received by a receiver and a receiver's shot successfully scores a goal. For this purpose, reward r depends on the results of a kicker's passing and a receiver's shooting. In preliminary experiments, we defined reward function $r(\sigma)$ given to episode σ, such as

$r(\sigma) = -30.0$ if $P_{\text{pass}}=\mathit{false}$,
$r(\sigma) = 5.0$ if $P_{\text{pass}}=\mathit{true}$ and $P_{\text{shot}}=\mathit{false}$,
and
$r(\sigma)=100.0$ if $P_{\text{pass}}=\mathit{true}$ and $P_{\text{shot}}=\mathit{true}$.

Proposition $P_{\text{pass}}/P_{\text{shot}}$ means that a kicker's pass/receiver's shot is successful. Identical reward $r(\sigma)$ is given to a kicker and a receiver by a trainer agent who judges whether a pass from a kicker to a receiver and the receiver's shot has succeeded. An episode consists of 50 simulation cycles, and takes five seconds in actual time. When a kicker or a receiver successfully scores a goal, the episode is concluded even before 50 simulation cycles have been completed.

4.6 Learning Experiments for a Kicker and a Receiver

We made four experiments to verify whether a kicker and a receiver can simultaneously learn their policies by applying the policy gradient method described in Section 3 to free-kick problems under the following conditions. A kicker and a receiver play against a defender and a goalie, as shown in Fig. 1. We exploited the programs of Trilearn Base [17] for the defender and goalie of the opponent. Trilearn Base is a program based on the UvA Trilearn2003 team's program that won the RoboCup2003 championship. Trilearn Base's defender and goalie are excellent players, while Trilearn2003's high level strategy is not implemented in Trilearn Base. Experiment 1 assumes that a kicker and a receiver have the same type of objective function, as shown in (1). However, they do not have interaction term $U_5(a_K,a_R;s)$, which Experiment 2 considers. In Experiment 3, a kicker has an objective function consisting of $U_1{}^K$, $U_2{}^K$, and $U_3{}^K$, and a receiver has an objective function consisting of $U_2{}^R$, $U_3{}^R$, and $U_4{}^R$. In Experiment 4, interaction term $U_5(a_K,a_R;s)$ was added to the objective functions used in Experiment 3. Experiments 3 and 4 deal with cases where a kicker and a receiver do not have heuristics $U_4{}^K$ and $U_1{}^R$ for deciding their next actions, respectively. Note that a receiver immediately makes a shoot if he can get the ball because we are only concentrating on making a receiver learn where to run to receive a pass, not how to dribble a ball.

5. EXPERIMENTAL RESULTS AND DISCUSSION

Four learning experiments described in Section 4.6 were conducted under conditions where ε=0.1 and T=10.0. Temperature parameter T is not lowered during the learning of ω^λ but fixed to search the parameter space for different good combinations of agents' policies. The initial values of weight ω^λ are selected at random from an interval between 10.0 and 30.0. Figures 2-4 show the changes of the passing success rate, the scoring rate, and the expectation of reward while learning 2000 episodes, respectively. Only their averages over every 50 episodes are plotted in the three figures. Changing the initial values of ω^λ, all four experiments were carried out ten times, and the values plotted in Figs. 2-4 are ensemble averages of the ten sets of experiments. Fig. 5 shows a typical example of the change of weight coefficients $\{\omega_i{}^\lambda\}$in the ten trials.

5.1 Experiment 1

In Experiment 1, a kicker and a receiver use the same type of objective functions consisting of four heuristics from $U_1{}^\lambda$ to $U_4{}^\lambda$. Figs. 5a and b shows that the ratio of the magnitudes of a kicker's $\{\omega_i{}^K\}$ approaches that of a receiver's $\{\omega_i{}^R\}$ as learning continues, even if the initial values of a kicker's $\{\omega_i{}^K\}$ are different from a receiver's $\{\omega_i{}^R\}$. The passing success rate increases from 42% to 78% in Fig. 2, and the scoring rate increases by 4 times from 8% to 32% in Fig. 3. These two increases mean that both the kicker and the receiver have acquired the same policy for action decisions and that successful passes contribute to scoring goals.

5.2 Experiment 2

In Experiment 2, interaction term U_5 is added to the objective function used in Experiment 1. Figs. 5c and 5d show the same tendency of $\{\omega_i{}^\lambda\}$ as in Experiment 1 except $\omega_5{}^\lambda$. The value of kicker's $\omega_5{}^K$ became much larger than receiver's $\omega_5{}^R$. This means that a kicker follows a receiver's action to realize cooperative play. As a result, the scoring rate is improved and becomes slightly larger than in Experiment 1 where there is no interaction term U_5.

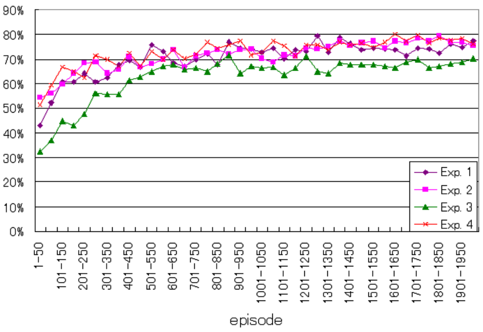


FIGURE 2: Passing Success Rate.

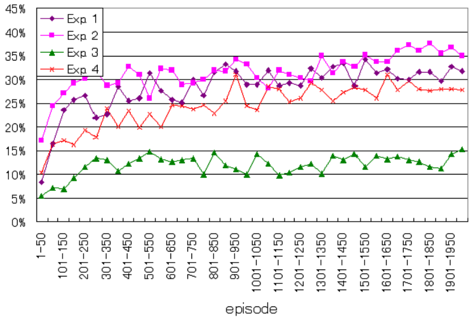


FIGURE 3: Scoring Rate.

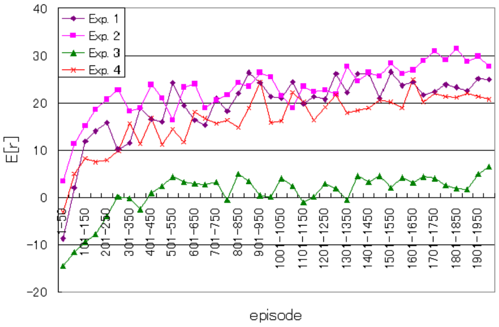


FIGURE 4: Expectation of Reward.

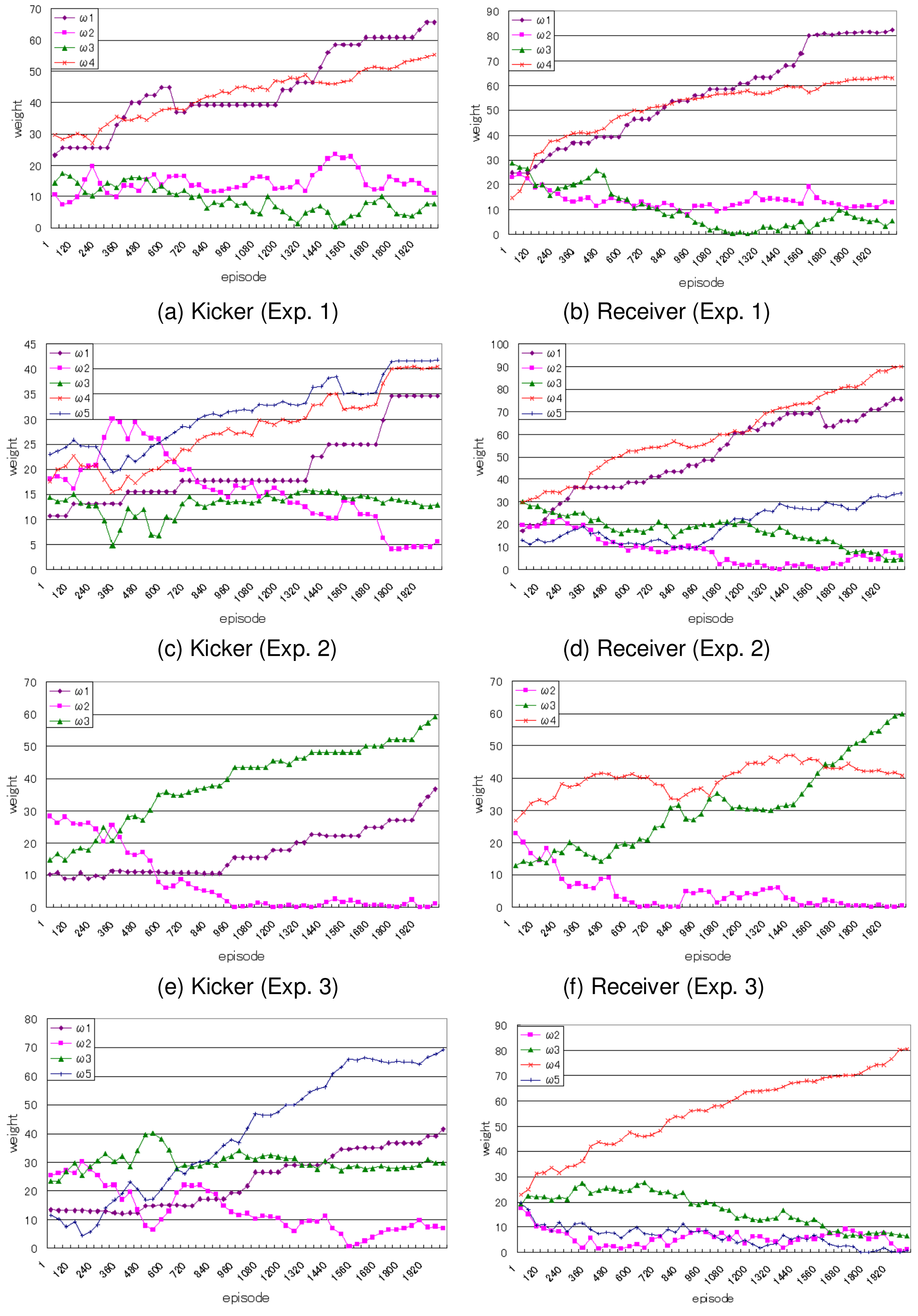
5.3 Experiment 3

In Experiment 3, a kicker has a set of heuristics, U_1^K, U_2^K, and U_3^K, which is different from a receiver's set of heuristics, U_2^R, U_3^R, and U_4^R. Passing success rate increases from 30% to 70%, as shown in Fig. 2, and the scoring rate increases by three times, from 5% to 15%, after learning 2000 episodes, as shown in Fig. 3. Figs. 5e and 5f show a common tendency on the changes of ω_2^λ and ω_3^λ for kickers and receivers, who realize cooperative play by making the common two heuristics, U_3^λ and U_4^λ, dominant in their policies. However, while watching their actual play in Experiment 3, we observed that their actions do not completely agree with each other. A small discrepancy exists between the cell to which a kicker passes the ball and the cell to which a receiver runs to receive it. This small discrepancy allows the defenders and the goalie time for defending. That is proved by the very low scoring rate shown in Fig. 3, whereas the passing success rate is not so bad when compared with other three experiments in Fig. 2.

5.4 Experiment 4

Interaction term U_5 is added to the policy used in Experiment 3. After learning 2000 episodes, the observed passing success rate is about 6 points higher than that in Experiment 3 (Fig. 2). Fig. 3 shows that the scoring rate increases about twofold the rate observed in Experiment 3. Moreover, expectation of reward E[r] increases by three times that obtained in Experiment 3 (Fig. 4).

In Figs. 5g and 5h, kicker's ω_5^K and receiver's ω_4^R become much larger than the other weight coefficients. Since ω_5^R nearly becomes zero after learning (Fig. 5h), a receiver does not try to predict a kicker's action. The receiver does not leave its current position, because ω_4^R is very large and the other weights are nearly zero. That is, the receiver stays and waits at his own current position. A kicker understands the receiver's intention and follows the receiver's choice. This kicker's policy was obtained by increasing weight ω_5^R of interaction term U_5 because a kicker does not have heuristics U_4^K that favors a pass close to a receiver. If the two agents do not have the same type of heuristics, the interaction term based on prediction of a teammate's decision model accelerates learning a master-servant relation between a kicker and a receiver, where a receiver is a master and a kicker is a servant as shown in Fig.6.



(a) Kicker (Exp. 1)

(b) Receiver (Exp. 1)

(c) Kicker (Exp. 2)

(d) Receiver (Exp. 2)

(e) Kicker (Exp. 3)

(f) Receiver (Exp. 3)

(g) Kicker (Exp. 4)

(h) Receiver (Exp. 4)

FIGURE 5: Example of Change of Weight Coefficients $\{\omega_i^{\lambda}\}$ during Learning Process in Experiments 1-4.

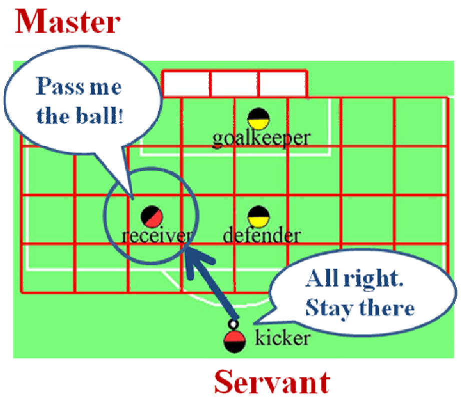


FIGURE 6: Master-servant Relation Obtained by the Interaction Term in Experiment 4, where a Kicker and a Receiver do not Have the Same Type of Heuristics.

6. CONSLUSION & FUTURE WORK

As an example of multi-agent learning problems, we considered a learning problem between a kicker and a receiver when a direct free kick is awarded just outside the opponent's penalty area in the soccer games of the RoboCup 2D Soccer Simulation League. We proposed a function that expresses heuristics for evaluating how target position is advantageous for safely sending/ receiving a pass and scoring a goal. This evaluation function does not depend on the dimension of the soccer field and the number of players. The weight coefficients in the function were learned by a policy gradient method. However, we did not try to make the weight coefficients converged to certain values, for example, by decreasing ε in learning rule (7). Our method needs criteria for deciding which weight coefficients should be best.

The heuristics includes an interaction term between a kicker and a receiver to intensify their coordination. The interaction term works to make an agent follow the teammate's action. To calculate the interaction term, we let a kicker/receiver agent have a receiver's/kicker's action-decision model to predict the teammate's action. Except for the interaction term, information on the action-decision model is exchanged with a teammate at a certain time interval, i.e, every 10 episodes during the learning process. The results of our learning experiments show that even if a kicker's and a receiver's heuristics are different, scoring rate is increased about two times of that obtained by learning without the interaction term. This means that adding an interaction term, which makes an agent follow a teammate's action, and predicting the teammate's action using the teammate's action-decision model are very effective for two agents to acquire a common policy, even if they do not completely have identical action-decision heuristics. However, our method needs human heuristics enough for solving the target problem, and the action-decision model of a teammate for predicting the teammate's action.

In the future, our agents will have and learn both passer and receiver policies so that they can learn wall-passes. An agent must switch its role from a receiver to a passer, and vice versa. We will also apply our method to more general cooperative play such as keeping the ball away from opponents and making through passes. Using action-decision models of teammates would be very useful for learning such team play.

7. REFERENCES

1. G. Weiss, S. Sen, editors. "*Adaption and Learning in Multi-agent Systems*", Springer-Verlag, Germany, 1996

2. S. Sen, G. Weiss, "*Learning in Multiagent Systems*". In G. Weiss, editor, "*Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*", pp. 259-208, The MIT Press, 1999

3. RoboCup 2D Soccer Simulation League, http://sourceforge.net/apps/mediawiki/sserver/index.php?title=Main_Page (access time: 14.03.2011)

4. S. Arai, K. Miyazaki. “*Learning Robust Policies for Uncertain and Stochastic Multi-agent Domains*”. In Proceedings of 7th International Symposium on Artificial Life and Robotics, pp.179-182, 2002

5. W.S. Lovejoy. “*A survey of algorithmic methods for partially observed Markov decision processes*”. Annals of Operations Research, 28(1): 47-65, 1991

6. R. S. Sutton, A. G. Barto.”*Reinforcement Learning*”, The MIT Press, 1998

7. L. P. Kaelbling, M. L. Littman and A. W. Moore. “*Reinforcement Learning: A Survey*”. Journal of Artificial Intelligence Research, 4:237-285, 1996

8. H. Kimura, K. Miyazaki and S. Kobayashi. “*Reinforcement Learning in POMDPs with Function Approximation*”. In Proceedings of the 14th International Conference on Machine Learning, pp. 152-160, 1997

9. T. Andou. “*Refinement of Soccer Agents' Positions Using Reinforcement Learning*”. In H. Kitano, editor, “*RoboCup-97: Robot Soccer World Cup I”*, pp. 373-388, Springer-Verlag, Berlin, 1998

10. M. Riedmiller, T. Gabel. “*On Experiences in a Complex and Competitive Gaming Domain – Reinforcement Learning Meets RoboCup–*”. In Proceedings of the 2007 IEEE Symposium on Computational Intelligence and Games (CIG2007), pp.17-23, 2007

11. P. Stone, G. Kuhlmann, M. E. Taylor and Y. Liu. “*Keepaway Soccer: From Machine Learning Testbed to Benchmark*”. In A. Bredenfeld, A. Jacoff, I. Noda and Y. Takahashi, editors, “*RoboCup 2005: Robot Soccer World Cup IX*”, pp. 93-105, Springer-Verlag, New York, 2006

12. S. Kalyanakrishnan, Y. Liu and P. Stone. “*Half Field Offense in RoboCup Soccer –A Multiagent Reinforcement Learning Case Study*”. In G. Lakemeyer, E. Sklar, D. G. Sorrenti, T. Takahashi, editors, “*RoboCup-2006: Robot Soccer World Cup X*”, pp.72-85, Springer-Verlag, Berlin, 2007

13. R. J. Williams. “*Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning*”. Machine Learning, 8(3-4): 229-256, 1992

14. H. Igarashi, S. Ishihara and M. Kimura. “*Reinforcement Learning in Non-Markov Decision Processes -Statistical Properties of Characteristic Eligibility-*”, IEICE Transactions on Information and Systems, J90-D(9):2271-2280, 2007 (in Japanese). This paper is translated into English and included in The Research Reports of Shibaura Institute of Technology, Natural Sciences and Engineering (ISSN 0386-3115), 52(2): 1-7, 2008

15. S. Ishihara, H. Igarashi. “*Applying the Policy Gradient Method to Behavior Learning in Multiagent Systems: The Pursuit Problem*”, Systems and Computers in Japan, 37(10):101-109, 2006

16. L. Peshkin, K. E. Kim, N. Meuleau and L. P. Kaelbling. “*Learning to cooperate via policy search*”. In Proceedings of 16th Conference on Uncertainty in Artificial Intelligence (UAI2000), pp. 489-496, 2000

17. J. R. Kok. “*UvA Trilearn*”, http://remote.science.uva.nl/~jellekok/robocup/ (access time: 05.01.2011)

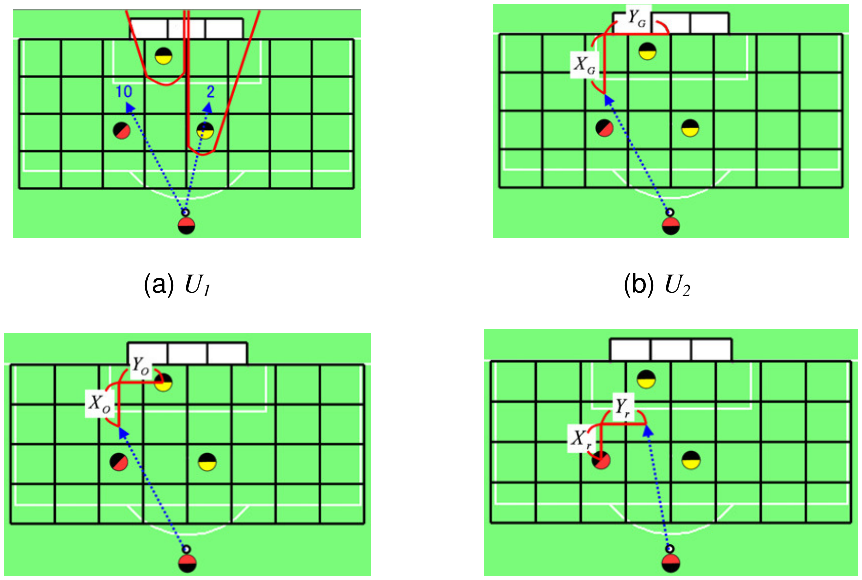
Appendix: Heuristic Functions from U_1^λ to U_4^λ

In our learning experiments, the following four heuristic functions $\{U_i^\lambda(a)\}$ $(i=1,2,3,4)$ were used to evaluate the suitability of the selected cell for kickers and receivers. In this section, let cell k be the one selected by kicker or receiver action a. All heuristic functions are normalized to avoid exceeding 10.0. Fig. 7 illustrates definitions of the four heuristic functions schematically.

(i) $U_1^\lambda(a;s)$: considers the existence of opponents in the pass direction and is defined by

$$U_1^\lambda(a;s) = \begin{cases} 2.0 & \text{if } \theta_{\text{p-opp}} \le 15^o \text{ and } d_{\text{k}} \ge d_{opp} - 2.0 \\ 10.0 & \text{else} \end{cases}, \tag{13}$$

where $\theta_{\text{p-opp}}$ is an angle between a pass direction and a direction to an opponent. d_{k} is the distance between the center of cell k and a kicker. d_{opp} is the distance between a kicker and the nearest opponent to the center of cell k.



(a) U_1

(b) U_2

(c) U_3

(d) U_4

FIGURE 7: Heuristics $\{U_i^\lambda\}$ $(i=1,2,3,4)$.

(ii) $U_2^\lambda(a;s)$: expresses a heuristics where shooting from nearer to the goal mouth has a greater chance of scoring a goal. It is defined by

$$U_2^\lambda(a;s) = \left(-|X_G| - |Y_G| + 37.5\right)/3.5 , \tag{14}$$

where vector $(X_{\text{G}}, Y_{\text{G}})$ is a distance vector between the center of cell k and the center of the goal mouth.

(iii) $U_3^\lambda(a;s)$: evaluates a distance between the center of cell k and the nearest opponent to the center of cell k as follows:

$$U_3^\lambda(a;s) = \left(|X_0| + |Y_0|\right)/5.0 , \tag{15}$$

where vector (X_0,Y_0) is a distance vector between the center of cell k and the position of the nearest opponent. Heuristics U_3 means that it is desirable to receive a pass near a place without opponents.

(iv) $U_4^{\lambda}(a;s)$: considers the distance between the center of cell k and the current receiver's position. It is defined by

$$U_4^{\lambda}\left(a;s\right)=\left(-\left|X_r\right|-\left|Y_r\right|+50.0\right)/5.0\ , \tag{16}$$

where vector $(X_{\mathrm{r}},Y_{\mathrm{r}})$ is a distance vector between the center of cell k and a receiver. If the distance is small, receivers can easily receive a pass.

Unsupervised Feature Selection Based on the Distribution of Features Attributed to Imbalanced Data Sets

Mina Alibeigi alibeigi@cse.shirazu.ac.ir
Computer Science and Engineering Department
Shiraz University
Shiraz, 71348-51154, Iran

Sattar Hashemi s_hashemi@shirazu.ac.ir
Computer Science and Engineering Department
Shiraz University
Shiraz, 71348-51154, Iran

Ali Hamzeh hamzeh@shirazu.ac.ir
Computer Science and Engineering Department
Shiraz University
Shiraz, 71348-51154, Iran

Abstract

Since dealing with high dimensional data is computationally complex and sometimes even intractable, recently several feature reduction methods have been developed to reduce the dimensionality of the data in order to simplify the calculation analysis in various applications such as text categorization, signal processing, image retrieval and gene expressions among many others. Among feature reduction techniques, feature selection is one of the most popular methods due to the preservation of the original meaning of features. However, most of the current feature selection methods do not have a good performance when fed on imbalanced data sets which are pervasive in real world applications.

In this paper, we propose a new unsupervised feature selection method attributed to imbalanced data sets, which will remove redundant features from the original feature space based on the distribution of features. To show the effectiveness of the proposed method, popular feature selection methods have been implemented and compared. Experimental results on the several imbalanced data sets, derived from UCI repository database, illustrate the effectiveness of the proposed method in comparison with other rival methods in terms of both AUC and F1 performance measures of 1-Nearest Neighbor and Naïve Bayes classifiers and the percent of the selected features.

Keywords: Feature, Feature Selection, Filter Approach, Imbalanced Data Sets.

1. INTRODUCTION

Since data mining is capable of finding new useful information from data sets, it has been widely applied in various domains such as pattern recognition, decision support systems, signal processing, financial forecasts and etc [1]. However by the appearance of the internet, data sets are getting larger and larger which may lead to traditional data mining and machine learning algorithms to do slowly and not efficiently. One of the key solutions to solve this problem is to reduce the amount of data by sampling methods [2], [3]. But in many applications, the number of instances in the data set is not too large, whereas the number of features in these data sets is more than one thousands or even more. In this case, sampling is not a good choice. Theoretically, having more features, the discrimination power will be higher in classification. However, this theory is not always true in reality since some features may be unimportant to predict the class labels or even be irrelevant [4], [5]. Since many factors such as the quality of the data, are responsible in the success of a learning algorithm, in order to extract information more efficiently, the data set should not contains irrelevant, noisy or redundant features [6]. Furthermore, high dimensionality of data set may cause the “curse of dimensionality” problem [7]. Feature reduction (dimensionality reduction) methods are one of the key solutions to all these problems.

Feature reduction refers to the problem of reducing the dimension by which the data set is described [8]. The general purpose of these methods is to represent data set with fewer features to reduce the computational complexity whereas preserving or even improving the discriminative capability [8]. Since feature reduction can brings a lot of advantages to learning algorithms, such as avoiding over-fitting and robustness in the presence of noise as well as higher accuracy, it has attracted a lot of attention in the three last decades. Therefore, vast variety of feature reduction methods suggested which are totally divided into two major categories including feature extraction and feature subset selection. Feature extraction techniques project data into a new reduced subspace in which the initial meaning of the features are not kept any more. Some of the well-known state-of-the-art feature extraction methods are principal component analysis (PCA) [5], non-linear PCA [13] and linear discriminant analysis (LDA) [13]. In comparison, feature selection methods preserve the primary information and meaning of features in the selected subset. The purpose of these schemes is to remove noisy and redundant features from the original feature subspace [13]. Therefore, due to preserving the initial meaning of features, feature selection approaches are in more of interest [8], [9].

Feature selection methods can be broadly divided into two categories: filter and wrapper approaches [9]. Filter approaches choose features from the original feature space according to pre-specified evaluation criterions, which are independent of specified learning algorithms.
Conversely, wrapper approaches select features with higher prediction performances estimated according to specified learning algorithms. Thus wrappers can achieve better performance than filters. However, wrapper approaches are less common than filter ones because they need higher computational resources and often intractable for large scale problems [9]. Due to their computational efficiency and independency to any specified learning algorithm, filter approaches are more popular and common for high dimensional data sets [9].

As was stated above, feature selection has been studied intensively [4], [5], [6], [8], [9] but its importance to resolving the class imbalance problem was recently mentioned by researchers [10]. The class imbalance problem refers to the issue that occurs when one or more classes of a data set have significantly more number of instances (majority class) than other classes of that data set (minority class) [10].In this type of data sets, the minority class has higher importance than the majority class. Since, nowadays, imbalanced data sets are pervasive in real world applications such as biological data analysis, text classification, web categorization, risk management, image classification, fraud detection and many other applications, it is important to propose a new feature selection method which is appropriate for imbalanced data sets.

Therefore, in this study, we present a new filter unsupervised feature selection algorithm which has the benefits of filter approaches and is designed to have a high performance on imbalanced data sets. The proposed approach chooses more informative features considering the importance of the minority class, according to relation between the distributions of features which are approximated by probability density function (PDF). The main idea of the proposed scheme is firstly approximating the PDF of each feature independently in an unsupervised manner and then removing those features for which their PDFs have higher covering areas with the PDFs of other features which are known as redundant features.

The rest of this paper is organized as follow. Section 2 discusses the related researches for unsupervised feature selection. Section 3 explains the proposed method for unsupervised feature selection applications. Our experimental results are given in section 4 and section 5 concludes the paper by a conclusion part.

2. RELATED WORK

Conventional feature selection methods evaluate various subsets of features and select the best subset among all with the best evaluation according to an effective criterion related to the application. These methods often suffer from high computational complexity through their searching process when applied to large data sets. The complexity of an exhaustive search is exponential in terms of the number of features of the data set. To overcome these shortcomings, several heuristic schemas have been proposed such as Branch and Bound (B&B) method which guarantees to find the optimal subset of features with computational time expectedly less than the exponential under the monotonicity assumption [12]. B&B starts from the full set of features and removes features by a depth first search strategy until the removing of one feature can improve the evaluation of the remaining subset of features [12]. Another popular approach is Sequential Forward Selection (SFS) which searches to find the best subset of features in an iterative manner starting from the empty set of features. In each step, SFS adds that feature to the current subset of selected features which yields to maximize the evaluation criterion for the new selected feature subset [13]. However, heuristic approaches are simple and fast with quadratic complexity, but they often suffer from lack of backtracking and thus act poorly for nonmonotonic criterions. In [24], another heuristic method called Sequential Floating Forward Selection (SFFS) was proposed which performs sequential forward selection with the backtracking capability at the cost of higher computational complexity.

The former methods can be applied in both supervised and unsupervised schemas according to their evaluation criteria. Since the interest of this paper is developing an unsupervised feature selection method, here, we investigate only the unsupervised methods. These methods can be generally divided into two divisions: filter and wrapper approaches [4], [8], [13]. The principle of wrapper approaches is to select subset of features regarding a specified clustering algorithm. These methods find a subset of features that using them for training a specified clustering; the highest performance can be achieved. Some examples of these approaches are [14], [15], [16]. Conversely, filter methods select features according to an evaluation criterion independent of specified clustering algorithm. The goal of these methods is to find irrelevant and redundant features and remove them from the original feature space. In order to find irrelevant and redundant features, various dependency measures have been suggested such as correlation coefficient [6], linear dependency [18] and consistency measures [19].

In this paper, we propose a feature subset selection based on the distribution of features which is able to handle the nonlinearity dependency between features in an unsupervised framework with a high performance for imbalanced data sets because of considering higher importance of the minority class which is the most important class in an imbalanced data set. The following section explains the proposed method in details.

3. THE PROPOSED UNSUPERVISED FEATURE SELECTION METHOD ATTRIBUTED TO IMBALANCED DATA SETS

The proposed unsupervised feature selection which is a filter approach attributed to imbalanced data sets, includes four steps. In the first step features are scaled in the range [0, 1]. Then, the probability density function (PDF) of each feature is estimated which gives a good overview about the distribution of instances for a specific feature. The third step is computing the number of times that the PDF of one feature is similar to PDF of other remaining features. At last, features with higher counter of being similar to other features are removed. Each step is described in details as follows.

The proposed method finds the relation between each two features as if they are similar or not according to their PDFs and removes those features which are more similar to other features as redundant features because all or most of their information is repeated in other features.

As was explained before, the first step in the proposed feature selection approach is scaling feature values in the range [0, 1]. Afterwards, PDF is estimated for each feature. The methods for estimating probability density functions can be totally categorized into parametric and non-parametric approaches [21]. The parametric methods assume a particular form for the density, such as Gaussian, so that only the parameters (mean and variance) need to be estimated. In comparison, non-parametric methods do not assume any knowledge about the density of the data and computes the density directly from the instances and because of this reason they are in more of interest. The general form of non-parametric probability density estimation methods is according to the following formula:

$$p(x) \cong \frac{k}{N*V} \qquad (1)$$

where, *p(x)* is the value of the estimated probability density function for instance *x*, *V* is the volume surrounding *x*, *N* is the total number of instances and *k* is the number of instances inside *V*. Two basic approaches can be adapted to practical non-parametric density estimation methods based on the status of *k* and *V*. Fixing the value of *k* and determining the corresponding volume *V* that contains exactly k instances inside, leads to methods commonly referred to as *K Nearest Neighbor* (*KNN*) methods. On the other hand, when the volume *V* is chosen to be fixed and *k* is determined, the non-parametric estimation method is called *Kernel Density Estimation* (*KDE*). Generally, the probability densities that estimated via *KNN* approaches are not very satisfactory because of some drawbacks. Because, *KNN PDF estimation* methods are prone to local noise. Moreover, the resulting PDF via KNN method is not a true probability density since its integral over all the instance space diverges [25]. In spite of these reasons, in this study, we estimate probability density functions through the *KDE* method with Gaussian kernel. It is noted that our proposed feature selection algorithm is not sensitive to any particular estimation method. However, using more accurate estimation methods cause the algorithm to perform more efficiently.

In order to compare PDFs of different features, all feature values are scaled into the [0, 1] interval because the range of various features may be different. Afterwards, the probability density functions for each of the features are computed according to *KDE* methods.

Having estimated the probability density function for each feature, the similarity between each of the two features is calculated. Two features are considered as similar features if the Mean Square Error (MSE) of their PDFs be less than a user specified threshold. Similar features contain nearly the same information because their PDFs are sufficiently similar. Thus, one of the similar features can be removed without a considerable loss of information. Among similar features, features which are similar to more other features of the whole feature space are removed. By removing the feature which has higher frequency of being similar with other features, the loss of information is minimized. Also, as the instances of all classes contribute equally for estimating the PDF of each feature, then instances of the minority classes are given higher importance in the PDF estimation process. Thus, features which are more informative according to minority classes are given higher chance to be selected. Algorithm 1 represents the steps of the proposed feature selection approach.

4. EXPERIMENTAL RESULTS AND DISCUSSION

The comparisons were carried out in three data sets coming from the UCI Machine Learning Repository including Ecoli, Ionosphere and Sonar which are all imbalanced. Table I shows a summary of the characteristics of the data sets used in this paper to assess the performance of the proposed method. The first column of Table I shows the name of the data set. Number of features and number of classes are showed in the second and third columns, respectively. The last column in each row is the number of instances per each class.

In order to evaluate the performance of a feature selection method, the performance of classifiers trained on the features selected by the mentioned feature selection method, is compared to the performance of classifiers trained on the full set of features named as baseline performance. There are many classifiers in machine learning domains with different biases. The most well-known classifiers for evaluating a feature selection method are *Naive Bayes* (*NB*) [11] classifier and *K- Nearest Neighbor* (*KNN*) classifier [13]. Naïve Bayes is a simple probabilistic classifier based on the assumption of class conditional independence of features [25]. K-Nearest Neighbor is a lazy learning algorithm which classifies each new test instance based on its *K* nearest training instances [25].

For imbalanced data sets, classifiers have difficulties to classify instances from the minority class because they simply classify instances as the majority class achieving a high accuracy. So, in these data sets, accuracy is not a good performance measure. There are a number of other statistics such as AUC (Area Under receiver operating characteristic Curve) and F-measures [26]. AUC and F1-measure are two of the statistics which are commonly used to evaluate classifiers focusing on the importance of the minority class. In this paper, we evaluate the performance of different feature selection methods based on AUC and F1-measure evaluation statistics.

Algorithm 1: The steps of the proposed unsupervised feature selection method.

```
Unsupervised Feature Selection Based on the Distribution of Features

Input: $D=\{d_1, d_2, \ldots, d_N\}$ // N is the number of instances
Input: $F=\{f_1, f_2, \ldots, f_n\}$   // n is the number of features
Output: $F^{(S)}$            // The selected subset of features, at the beginning, $F^{(S)} = F$

Begin

Step 1. Scale each feature in range [0,1]
Step 2. Estimate the probability density function (PDF) for each feature
Step 3. For  i=1 to n-1
              For  j=i+1 to n
                   Calculate MES(density of feature i, density of feature j)
             If MSE <= epsilon
                   Consider features i and j to be similar
                   Increment the similarity counter of both features i and j
Step 4. Between each similar features, remove feature with higher similarity counter
Step 5. Return list of remaining features as the list of selected features ($F^{(S)}$)

End
```

Name	# Features	# Class	# Instances Per Class
Sonar	60	2	97, 111
Ionosphere	34	2	126, 225
Ecoli	7	8	143, 77, 52, 35, 20, 5, 2, 2

TABLE 1: Characteristics of data sets used in this study for experimental evaluations.

Comparisons are done in Weka framework [22]. To show the effectiveness of the proposed method, we compared our method with two of the commonly used supervised approaches proposed by Hall et al. [19] and Lie et al. [4] named as Correlation-based Feature subset Evaluation and Consistency-based feature Subset Evaluation, which are abbreviated in results as CfsSubsetEval and ConsistencySubsetEval, respectively. We also compared the proposed method with an unsupervised Sequential Forward Selection (SFS) scheme for which Entropy is used as the evaluation criterion. This method is illustrated as SFS with entropy in experiments. The entropy criterion for this method is defined according to formula (2).

$$Entropy = -\sum_{p=1}^{l}\sum_{q=1}^{l} \begin{array}{l}(sim(p,q)*\log(sim(p,q))+\\ (1-sim(p,q))*\log(1-sim(p,q))\end{array}$$

$$Sim(p,q) = e^{-\alpha D_{pq}} \qquad (2)$$

$$D_{pq} = [\sum_{j=1}^{M}(\frac{x_{p,j}-x_{q,j}}{\max_j - \min_j})^2]^{1/2}$$

where D_{pq} is the distance between two instances *p* and *q* and $x_{p,j}$ denotes the jth feature value for *instance p*. max_j and min_j are respectively the maximum and minimum values for the *j*th feature and *M* denotes the number of features. In (2), α is a positive constant which is set as $\alpha = \frac{-\ln 0.5}{\overline{D}}$ where $\overline{D}$ is the average distance between all instances.

Tables 2-4, separately illustrate the experimental results on each of the introduced data sets. The fist column of these tables, is the name of the feature selection method. The second column of each table, is the number of selected features by the corresponding feature selection method. AUC and F1 performance of Naïve Bayes (NB) classifier are shown in third and forth columns, respectively. Also, AUC and F1 performance of K-Nearest Neighbor (KNN) classifier are shown in the last two columns of each table.

As the results show in Tables 2-4, the AUC and F1 performance of the proposed method is fairly comparable to the performance of the Baseline method while the proposed method removes some redundant features (about half of the original features, see Figure 1) which lead to less computational complexity. This illustration acknowledges that feature selection is a key solution for classifiers on high dimensional imbalanced data sets.

Also, the proposed feature selection method has higher 1-NN classifier performance than CfsSubsetEval and ConsistencySubsetEval feature selection schemes in terms of both AUC and F1 evaluation measures and is comparable to both mentioned feature selection methods in terms of AUC and F1 performance of NB classifier. However, it is noticeable that the proposed method is an unsupervised approach which has access to less information in comparison with CfsSubsetEval and ConsistencySubsetEval feature selection methods which are supervised methods and have access to the class labels. Furthermore, our method has higher performance in comparison with other rival unsupervised feature selection scheme named as SFS with Entropy in experiments in terms of both AUC and F1 performances of 1-NN and NB classifiers. In general, it can be concluded that the proposed feature selection approach is more efficient than the other rival unsupervised feature selection and is comparable to the commonly used supervised feature selection schemas considered in experiments.

Figure 1 shows the comparison among rival feature selection methods in terms of the percent of selected features for each data set. Those feature selection methods which select a small percent of features while having a suitable performance, are more of interest. As can be seen, this property is true for the proposed feature selection method which makes it a good choice of action for feature selection on imbalanced data sets.

Feature Selection Method	# Selected Features	NB F1	NB AUC	1-NN F1	1-NN AUC
Baseline	34	0.829	0.935	0.857	0.822
CfsSubsetEval	14	0.92	0.958	0.885	0.852
ConsistencySubsetEval	7	0.872	0.926	0.875	0.849
SFS with Entropy	14	0.778	0.82	0.791	0.747
The Proposed Method	12	0.92	0.958	0.91	0.889

TABLE 2: Experimental results on Ionosphere data set in terms of the number of selected features and the AUC and F1 evaluation performances for NB and 1-NN classifiers.

Feature Selection Method	# Selected Features	NB F1	NB AUC	1-NN F1	1-NN AUC
Baseline	7	0.854	0.96	0.801	0.875
CfsSubsetEval	6	0.854	0.96	0.799	0.873
ConsistencySubsetEval	6	0.854	0.96	0.799	0.873
SFS with Entropy	6	0.791	0.947	0.766	0.854
The Proposed Method	6	0.854	0.96	0.799	0.873

TABLE 3: Experimental results on Ecoli data set in terms of the number of selected features and the AUC and F1 evaluation performances for NB and 1-NN classifiers.

Feature Selection Method	# Selected Features	NB F1	NB AUC	1-NN F1	1-NN AUC
Baseline	60	0.673	0.8	0.865	0.862
CfsSubsetEval	19	0.675	0.812	0.836	0.834
ConsistencySubsetEval	14	0.666	0.811	0.85	0.847
SFS with Entropy	14	0.557	0.658	0.659	0.657
The Proposed Method	14	0.652	0.769	0.88	0.878

TABLE 4: Experimental results on Sonar data set in terms of the number of selected features and the AUC and F1 evaluation performances for NB and 1-NN classifiers.

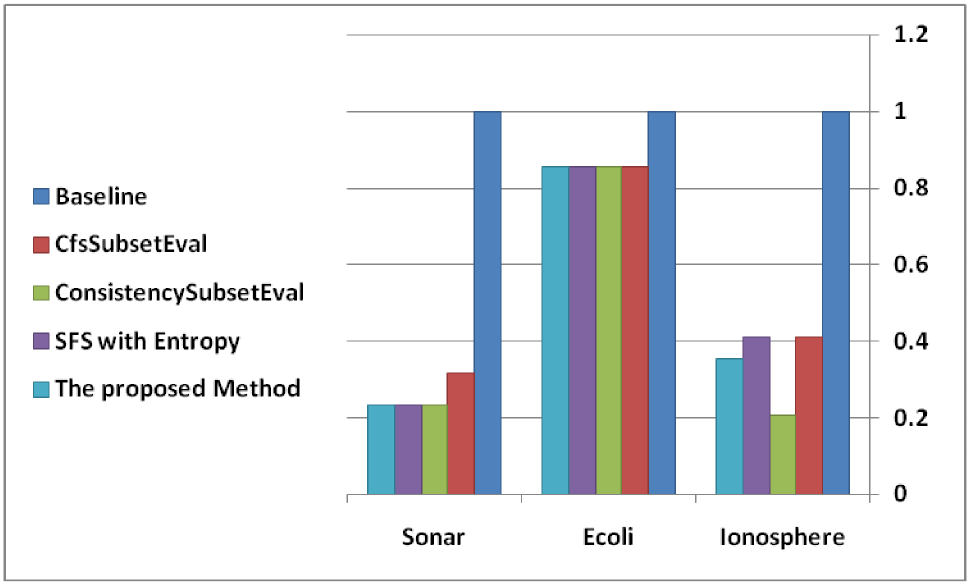


FIGURE 1: The percentage of selected features for each data set and in average for all data sets for the proposed feature selection method and other rival methods.

5. CONSLUSION AND FUTURE WORK

Feature selection techniques have a key role when encountering high dimensional data sets. Recently, filter based feature selection methods are of more interest because of their independence to any particular learning algorithm and their efficiency on high dimensional data sets. Since, most of the current feature selection methods perform poorly when fed on with imbalanced data sets, designing a feature selection method which is able to handle imbalanced data sets is recently mentioned by researchers.

Therefore, in this study, we proposed a new filter unsupervised feature selection scheme attributed to imbalanced data sets, which selects features based on the relation between probability density estimations of features. The main idea is that a feature, for which its distribution is more similar to the distribution of other features, is redundant because all or most of its information is repeated in those similar features. So, this feature can be removed from the original feature space with the least loss of information. Experimental results on a set of imbalanced data sets show that the proposed feature selection approach compared to the rival unsupervised feature selection method, can find a more informative subset of features which are more useful for classifying the instances of the minority classes. Also, the performance of the proposed method is comparable to the performance of two commonly used supervised feature selection frameworks in terms of both AUC and F1 evaluation measures.

For future work, it might be useful to apply this idea in the field of supervised feature selection and find the probability density functions per classes for each feature and find the similarity between features by considering their densities per classes.

6. ACKNOWLEDGEMENT

This work is supported by the Iran Tele Communication Research Center.

7. REFERENCES

1. U. Fayyad, G. Piatetsky-Shapiro and P. Smyth. *"From data mining to knowledge discovery in databases"*, AI Magazine, vol. 17, pp. 37–54, 1996

2. M. Lindenbaum, S. Markovitch and D. Rusakov. *"Selective sampling for nearest neighbor classifiers"*, Machine learning, vol. 54, pp. 125–152, 2004

3. A.I. Schein and L.H. Ungar, *"Active learning for logistic regression: an evaluation"*, Machine Learning, vol. 68, pp. 235–265, 2007

4. M.A. Hall. *"Correlation-based feature subset selection for machine learning"*, Ph.D. Dissertation, Department of Computer Science, University of Waikato, Hamilton, New Zealand, 1999

5. I.K. Fodor. *"A survey of dimension reduction techniques"*, Technical Report UCRL- ID-148494, Lawrence Livermore National Laboratory, US Department of Energy, 2002

6. M.A. Hall. *"Correlation-based feature selection for discrete and numeric class machine learning"*, Department of Computer Science, University of Waikato, Hamilton, New Zealand, 2000

7. R. Bellman. *"Adaptive Control Processes: A Guided Tour"*, Princeton University Press, Princeton, 1961
8. H. Liu, J. Sun, L. Liu and H. Zhang, *"Feature selection with dynamic mutual information"*, Pattern Recognition, vol. 42, pp. 1330 – 1339, 2009

9. N. Pradhananga. *"Effective Linear-Time Feature Selection"*, Department of Computer Science, University of Waikato, Hamilton, New Zealand, 2007

10. M. Wasikowski and X. Chen. *“Combating the small sample class imbalance problem using feature selection”*, IEEE Transactions on knowledge and data engineering, 2009

11. G.H. John and P. Langley. *“Estimating Continuous Distributions in Bayesian Classifiers*”. In: Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, pp. 338-345, 1995
12. M.P. Narendra and K. Fukunaga. *“A branch and bound algorithm for feature subset selection”*, IEEE Trans. Comput. Vol. 26, pp. 917–922, 1997

13. P.A. Devijver and J. Kittler. *“Pattern Recognition: A Statistical Approach”*, Englewood Cliffs: Prentice Hall, 1982

14. M. Dash and H. Liu. *“Unsupervised Feature Selection”*, Proc. Pacific Asia conf. Knowledge Discovery and Data Mining, pp. 110-121, 2000

15. J. Dy and C. Btodley. *“Feature Subset Selection and Order Identification for Unsupervised Learning”*, Proc. 17th Int’l. Conf. Machine Learning, 2000

16. S.Basu, C.A. Micchelli and P. Olsen. *“Maximum Entropy and Maximum Likelihood Criteria for Feature Selection from Multi-variate Data”*, Proc. IEEE Int’l. Symp. Circuits and Systems, pp. 267-270, 2000

17. S.K .Pal, R.K. De and J. Basak. *“Unsupervised Feature Evaluation: A Neuro-Fuzzy Approach”*, IEEE Trans. Neural Network, vol. 11, pp. 366-376, 2000

18. S.K .Das. *“Feature Selection with a Linear Dependence Measure”*, IEEE Trans. Computers, pp. 1106-1109, 1971

19. G.T. Toussaint and T.R. Vilmansen. *“Comments on Feature Selection with a Linear Dependence Measure”*, IEEE Trans. Computers, 408, 1972

20. H. Liu and R. Setiono. *“A probabilistic approach to feature selection - A filter solution”*. In: 13th International Conference on Machine Learning, pp. 319-327, 1996

21. K. Fukunaga. *“Introduction to Statistical Pattern Recognition”*, Academic Press, 2nd Ed. 1990

22. E. Frank, M.A. Hall, G. Holmes, R. Kirkby and B. Pfahringer. *“Weka - a machine learning workbench for data mining”*, In The Data Mining and Knowledge Discovery Handbook, pp. 1305-1314, 2005

23. M. Dash and H. Liu. *“Unsupervised Feature Selection”*, Proc. Pacific Asia Conf. Knowledge Discovery and Data Mining, pp. 110-121, 2000

24. P. Pudil, J. Novovicova and J. Kittler. *“Floating Search Methods in Feature Selection”*, Pattern Recognition Letters, vol. 15, pp. 1119-1125, 1994

25. R.O. Duda, P.E. Hart and D.G. Stork. *“Pattern Classification”*, Second Edition, Wiley, 1997

26. G. Forman. "An extensive empirical study of feature selection metrics for text classification", Journal of Machine Learning Research, vol. 3, pp. 1289-1305, 2003

INSTRUCTIONS TO CONTRIBUTORS

The main aim of International Journal of Artificial Intelligence and Expert Systems (IJAE) is to provide a platform to AI & Expert Systems (ES) scientists and professionals to share their research and report new advances in the field of AI and ES. IJAE is a refereed journal producing well-written original research articles and studies, high quality papers as well as state-of-the-art surveys related to AI and ES. By establishing an effective channel of communication between theoretical researchers and practitioners, IJAE provides necessary support to practitioners in the design and development of intelligent and expert systems, and the difficulties faced by the practitioners in using the theoretical results provide feedback to the theoreticians to revalidate their models. IJAE thus meets the demand of both theoretical and applied researchers in artificial intelligence, soft computing and expert systems.

IJAE is a broad journal covering all branches of Artificial Intelligence and Expert Systems and its application in the topics including but not limited to technology & computing, fuzzy logic, expert systems, neural networks, reasoning and evolution, automatic control, mechatronics, robotics, web intelligence applications, heuristic and AI planning strategies and tools, computational theories of learning, intelligent system architectures.

To build its International reputation, we are disseminating the publication information through Google Books, Google Scholar, Directory of Open Access Journals (DOAJ), Open J Gate, ScientificCommons, Docstoc and many more. Our International Editors are working on establishing ISI listing and a good impact factor for IJAE.

The initial efforts helped to shape the editorial policy and to sharpen the focus of the journal. Starting with volume 2, 2011, IJAE appears in more focused issues. Besides normal publications, IJAE intend to organized special issues on more focused topics. Each special issue will have a designated editor (editors) – either member of the editorial board or another recognized specialist in the respective field.

We are open to contributions, proposals for any topic as well as for editors and reviewers. We understand that it is through the effort of volunteers that CSC Journals continues to grow and flourish.

LIST OF TOPICS

The realm of International Journal of Artificial Intelligence and Expert Systems(IJAE) extends, but not limited, to the following:

- AI for Web Intelligence Applications
- AI in Bioinformatics
- AI Parallel Processing Tools
- AI Tools for CAD and VLSI Analysis/Design/Testing
- AI Tools for Computer Vision and Speech Understand
- AI Tools for Multimedia
- Application in VLSI Algorithms and Mobile Communic
- Automated Reasoning
- Case-based reasoning
- Data and Web Mining
- Derivative-free Optimisation Algorithms
- Emotional Intelligence
- Evolutionary and Swarm Algorithms
- Expert System Development Stages
- Expert Systems Components
- Expert-System Development Lifecycle
- Fuzzy Sets and logic
- Heuristic and AI Planning Strategies and Tools
- Hybridisation of Intelligent Models/algorithms
- Image Understanding
- Inference
- Integrated/Hybrid AI Approaches
- Intelligent Planning
- Intelligent Search

- Intelligent System Architectures
- Knowledge-Based Systems
- Logic Programming
- Multi-agent Systems
- Neural Networks for AI
- Parallel and Distributed Realisation of Intelligen
- Reasoning and Evolution of Knowledge Bases
- Rule-Based Systems
- Uncertainty
- Knowledge Acquisition
- Knowledge-Based/Expert Systems
- Machine learning
- Neural Computing
- Object-Oriented Programming for AI
- Problem solving Methods
- Rough Sets
- Self-Healing and Autonomous Systems
- Visual/linguistic Perception

CALL FOR PAPERS

Volume: 2 - **Issue:** 3 - May 2011

i. Paper Submission: May 31, 2011

ii. Author Notification: July 01, 2011

iii. Issue Publication: July /August 2011

CONTACT INFORMATION

Computer Science Journals Sdn BhD
M-3-19, Plaza Damas Sri Hartamas
50480, Kuala Lumpur MALAYSIA

Phone: 006 03 6207 1607
006 03 2782 6991

Fax: 006 03 6207 1697

Email: cscpress@cscjournals.org

CSC PUBLISHERS © 2011
COMPUTER SCIENCE JOURNALS SDN BHD
M-3-19, PLAZA DAMAS
SRI HARTAMAS
50480, KUALA LUMPUR
MALAYSIA

PHONE: 006 03 6207 1607
006 03 2782 6991

FAX: 006 03 6207 1697
EMAIL: cscpress@cscjournals.org