# A Task Distribution Based Q-Learning Algorithm for Multi- Agent Team Coordination

**Qiao Sun[1*], Zhibo Chen[1], Feixiang Chen[1], Fu Xu[1], Yanan Shi[2]**

[1] *School of Information Science and Technology, Beijing Forestry University, Beijing 100083, China*

[2] *College of Computer Science and Technology, Jilin University, Changchun 130012, China*

*Received 1 March 2014, www.cmnt.lv*

**Abstract**

It is difficult to apply traditional Q-learning algorithm to Multi-Agent environment, because in this case, the size of state-action space is so huge that it is hard to obtain the global optimal solution. In the paper, a task distribution based Q-learning algorithm is proposed to solve this problem. In this algorithm, at each learning step, it first distributes sub-task to each Agent dynamically. The Learning processes include the learning of task-distribution strategy and the learning of action-selection strategy synchronously, and every Agent shares the Q value table. Both Theoretical analysis and experimental results demonstrate that the proposed algorithm outperforms conventional Q-learning algorithm.

*Keywords*: Machine learning, Q-learning, Multi-Agent System, State-action space

## 1 Introduction

Reinforcement Learning (RL) is an important machine learning method, and Q-learning algorithm [1] is the most popular and efficient model-independent RL algorithm, which is essentially an asynchronous dynamic programming method based on Markov decision process [1], and it has been one of the core technologies for construction of Agent. It learns the optimal action strategy of dynamic system by sensing environmental conditions, selecting the appropriate action and obtaining uncertain reward value from environment [2]. RL has solved the problem of optimal action strategy of a single Agent in MDP environment [3].

Multi-Agent systems consist of many Agents which could autonomously run, complete complex tasks and address intricate problems by collaboration between Agents. In recent years, many researchers carried out researches in this field. In 2000, Cai et.al [4] proposed an multi-Agent RL model based on a single Agent RL algorithm, its feature is the introduction of a leading Agent as a protagonist of team learning, completing the learning of team by transformation of the role of leading Agent. In 2011, Wang et.al [5] introduced punishment operator and combined multi-satellite punishment operator to improve the original satellite Agent utility value gain function. On this basis, a multi-satellite Agent RL algorithm in order to solve multi-satellite cooperation task allocation strategy was proposed. Besides simulation experiment and analysis, the algorithm was proved that it could effectively solve the problem of multi-satellite cooperation task allocation. What's more, in 2010, Liu et.al [6]proposed a vote based multi-agent RL method to make the team learn to collaborate in the game. First by defining combination action called strategy to transmit collaborative problem into learning for strategy, then dividing the court to effectively reduce dimensions of state space. Thirdly distinguish the environment state and only consider collaboration state; combining individual player decisions by voting to achieve the purpose of collaboration. In 2010, Tang et.al [7] introduced the local information exchange item which has the function of diffusion based on reaction diffusion idea of multi-Agent systems; and utilized the performance potential theory, built a learning algorithm in order to solve the both unsynchronized moment and multi-site cooperative control strategy. In 2008, Xiao et.al [8] studied the behaviour of a rational conservative Agent, in the absence of any other information on the conditions of Agent, and a mixed strategy Nash equilibrium can be obtained. They proposed a regret-based conflict game RL model in Multi-Agent complex environment. The model established the process of belief updating, optimized the action selection strategy of conflict game by the introduction of cross-entropy distance. Moreover, in 2010, Wang et.al [9] introduced property maintenance operator, extended the classical belief model, and gave the rational maintain process of consciousness property, made Agent have the ability to conduct online learning in a dynamic environment, furthermore, based on models of consciousness they proposed an Agent architecture of adaptable and social nature, and based on this they also developed a path planning Agent.

Q learning algorithm [1] is the most popular efficient model-independent RL algorithm, by sampling of the objective world to learn the optimal action strategy. When traditional Q learning algorithm for single Agent is applied to the multi-Agent environment, due to the size of the state action space grows exponentially, it often takes a long time to converge [14]. To reduce the scale of state - action space, in general, researchers utilize divided independent

* ***Corresponding author's*** e-mail: sunqiao19800608@163.com.

RL method, but it is difficult to learn the optimal strategy from global angle [15], one of the reasons is that traditional method dose not divide tasks for each Agent, or just before the start of the learning divides tasks for Agent team according to a fixed strategy. Allocating a fixed sub-task to every Agent, during the learning process of each Agent the sub-task is always the same, so that each Agent in the learning process never considered the team's overall interests, learning results are only locally optimal strategy for each Agent respective sub-tasks, learning outcomes cannot adapt to the dynamic changes in the environment [16].

In this article we presented a task distribution based Q learning algorithm for Multi-Agent environment, it dynamically divided tasks for Agent team before every action selection step. It distributed tasks to each Agent and then each Agent selected corresponding action according to the allocated task. The proposed algorithm divided each learning step into two parts, one is to learn tasks division strategy, and the other is to learn action selection strategy of Agent for the completion of the respective sub-task.

## 2 Background of Q-Learning Algorithm

Reinforcement learning is a supervised learning [10-13] algorithm that is focus on how to make Agent perceive and act in one environment and select the optimal sequence of actions to achieve its goals. Every movement of Agent in the environment would cause the trainer get rewards or punishment. So Agent can learn to choose a series of actions from indirectly delayed reward in order to obtain the cumulative maximum reward.

Q learning algorithm [1] is the most popular efficient model-independent RL algorithm, by sampling of the objective world to learn the optimal action strategies. Defining Q (*s*, *a*) as the largest discount cumulative reward that Agent applied action *a* which is the first action can get from the state *s*. Thus, the optimal strategy of Agent is to choose the action which could make Q (*s*, *a*) value become the largest under each state. To learn Q function, Agent repeatedly observes its current state s, selects an action *a*, performs this action, and then observes the result return *r* = *r* (*s*, *a*) as well as new state *s*'. Then Agent follows each such transformation and updates Q (*s*, *a*) according to the following rule:

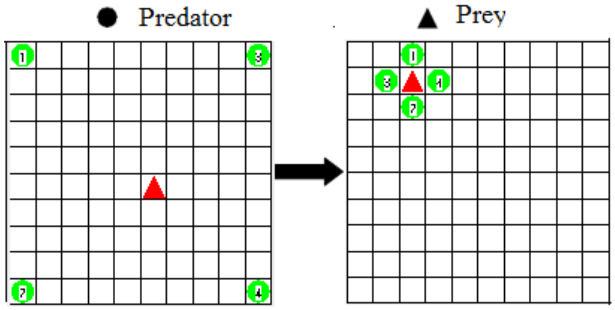
$$Q(s,a) \leftarrow r + \gamma \max_{a'} Q(s',a') \tag{1}$$

Q learning algorithm is to constantly correct action policy obtained through experience of trial and error in order to get the maximum return on action strategies. Experience is limited to the current observed state, action and the rewards. In the initial stages of learning, when Agent with no experience would completely rely on trial and error, as the study progresses, experience gradually accumulates, which inevitably would benefit for improving the action strategy.

## 3 The Proposed Method Based on Task distribution Strategy and Action Selection Strategy

### 3.1 DESCRIPTION OF THE PROBLEM DOMAIN

As shown in Figure 1, in n×n grid world, there are four hunters and a prey. A collaborative team contains four hunter Agents, and the hunter's goal is to eventually capture prey, the hunter can be viewed as predator. Every hunter and prey can at most only move one step, there are a total of five possible moving direction, namely east, west, south, north, stationary. Two Agent cannot occupy the same square, four hunters only simultaneously occupy four adjacent squares of prey can they capture prey successfully. Hunter has a perception radius, only prey in perception within the range of hunter can hunter perceive prey, which can be understood as the vision of hunter, hunter's vision has a certain range, he can only see something which the distance between he and it is in a certain range. In order to study collaborative multi-Agent team learning process, we regard prey of game Agent as a part of the environment. Hence, we can see that in order to achieve team goals hunters must collaborate with others.




(a) Initial State (b) Terminate State

FIGURE 1. Description of predator-prey pursuit game

### 3.2 TASK DISTRIBUTION LEARNING

The condition for successfully capturing prey is that the four hunters simultaneously take over four adjacent locations of the prey, and the traditional method has not division of tasks and direct learning capture action, or only once dividing tasks before the study begins. The former because each Agent dose not coordinate on task distribution, so has much blindness, and slow convergence. The latter is to divide overall destination capture prey to four sub-destinations, namely take over the east, south, west and north sides of the location of prey. Four hunters need to complete their sub-destinations in order to achieve the overall destination. Then just before the study begin assigning to each hunter Agent a fixed sub-destination. In the process of capturing Agent always captures for this sub-destination. Under this approach, if the hunter (hereinafter referred to as Hunter A) whose sub-destination is take over west location of prey went to south location of prey, the hunter (hereinafter referred to as Hunter B) whose sub-destination is take over north location of prey went to west location of prey, therefore they could capture as the original sub-destination, although a single member A or B could obtain local optimal solution, but apparently this is not the global optimum capture strategy. If at that time sub-destination of hunter A changes to north location of pray, sub-destinations of hunter B changes to west location of

prey, then capturing continues, obviously prey would be captured faster.

In this paper, we utilized a method which dynamically allocates sub-destinations, each hunter Agent before every move would re-allocate sub-destination, then captures for the new sub-destination, and the sub-destination allocation strategy is got through RL. To reduce state - action space, utilizing the relative position of the hunter Agent and prey to indicate the status of the game, stipulated to establish a coordinate system in two-dimensional location, direction of the longitudinal y-axis is from north to south, direction of the horizontal x-axis is the from west to east. The side length of a small location is equivalent to a unit length of coordinate system. When the hunter perceives prey, the relative position of the hunter and prey is expressed as two-tuple ($X_{Predator}$ - $X_{prey}$, $Y_{Predator}$ - $y_{prey}$), where ($X_{Predator}$, $Y_{Predator}$) represents the current position coordinate value of hunters in the location world, and ($X_{prey}$, $Y_{prey}$) indicates the current position coordinate value of prey, as a sub-destination cannot be given to two Agents at the same time, we utilize a 4-bit binary number mark as an orderly symbol of the current distribution of east, south, west, north four sub-destinations, such as 1101 = 13 represents three sub-destinations such as east, south and north have been assigned to other Agents, and the sub-destination west is not allocated, so the scope of this flag is 0000-1111, which corresponds to the decimal number 0-15, this flag can be expressed as an integer. Hence in the tasks division strategy learning, the status of each Agent can be expressed as $S_1$ = {$X_{Predator}$-$X_{prey}$, $Y_{Predator}$-$Y_{prey}$, flag}, action space is selection for four sub-target, expressed as $a_1$ = {east, south, west, north}, Q value updating rule of each Agent is

$$Q_1(S_1, a_1) = r_1 + \gamma \max_{a_1'} Q_1(S_1', a_1') \tag{2}$$

Among them r1 is the return value, for each hunter, if according to the sub-destination of this task distribution which has been allocated, then this allocated is unreasonable, hunter gets a negative return, otherwise utilizing the distance between hunters and sub-destinations $(X_{Predator}-X_{subdestination})^2+(Y_{Predator}-Y_{subdestination})^2$ to measure the task distribution strategy is good or not, if the distance between the hunter and the allocated sub-destination is in a smaller value in the distance between hunter and any other sub-destination, namely hunter and the allocated sub-destination is closer than the other sub-destination, then this task distribution was given a large positive return value, otherwise this task distribution gives small positive return value. Because action strategies of team members can be shared, so each member Agent would share the Q value, thereby improving the learning efficiency.

## 3.3 ACTION SELECTION LEARNING

For every hunter Agent, the purpose of this stage is to learn a series of actions strategy, by performing the action, and the Agent can complete your own sub-destination. Every possible movement direction of Hunter Agent is east, south, west, north, and stationary. Utilizing relative coordinates of hunters and their sub-destinations to represent the state, then the state is represented as

$S_2$={$X_{Predator}$ - $X_{Sub\text{-}destination}$, $Y_{Predator}$ - $Y_{Sub\text{-}destination}$}.

| | | | | |
|---|---|---|---|---|
| (-4,-2) | (-3,-2) | (-1,-2) | (0,-2) | (1,-2) |
| (-4,-1) | (-3,-1) | (-1,-1) | (0,-1) | (1,-1) |
| (-4,0) | (-3,0) | (-1,0) | Sub-destination | Prey |
| (-4,1) | (-3,1) | (-1,1) | (0,1) | (1,1) |
| (-4,2) | (-3,2) | (-1,2) | (0,2) | (1,2) |

FIGURE 2. State analysis chart of capture action selection

Figure 2 shows when the hunter perception radius is 2, sub-destination of hunter is to occupy the west square of prey, obtained different state value of different positions of hunter. Action space is $a_2$ = {east, south, west, north, and stationary}, Q value update rule for each Agent is

$$Q_2(S_2, a_2) = r_2 + \gamma \max_{a_2'} Q_2(S_2', a_2') \tag{3}$$

Among them $r_2$ is return value, when hunter chooses an action and obtains the highest return when reaches his sub-destination, if shortening the distance between hunter and sub-destination, then obtains the second highest return, if no change in the distance between hunter and sub-destination, then obtains return 0, if the distance with prey increases, then obtains negative return. And different hunters' Q function could be shared to improve learning efficiency.

## 3.4 FLOW CHART OF THE PROPOSED ALGORITHM

Through the above description, the specific algorithm of each hunter Agent is as follows:

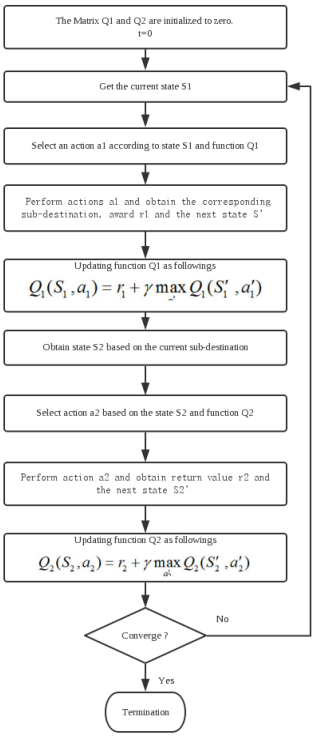
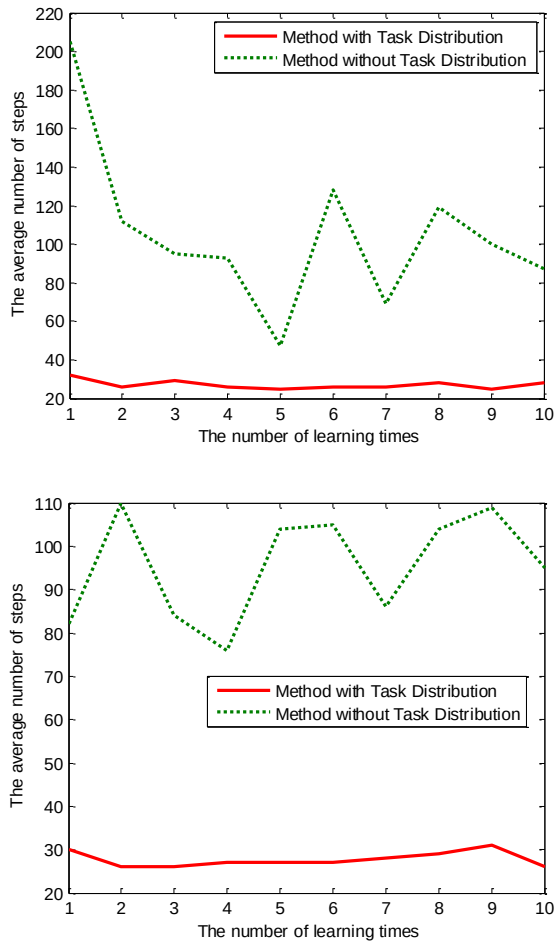



FIGURE 3. Flow Chart of the proposed algorithm

## 4 The Experimental Results

For predator-prey pursuit game, we utilize two-dimensional 10×10 grids, four predators are numbered as 0,1,2,3. Hunter and prey are limited in the 10×10 grids. Initial position of four predators are located at (0,0), (0,9), (9,0), (9,9). Prey is in the (5, 5). We carry out the experiment to compare the proposed method with task distribution dynamically and the traditional method without task respectively, each experiment make 1000 times capture, and we record the number of steps required for each capture, taking every 50 times capture as a group and calculate the average of the number of steps. There are 10 groups of data in total.

Figure 4 shows the comparison of the proposed method with task distribution and the traditional method without task distribution, the horizontal axis of the figure represents the number of learning times, and the ordinate represents the average number of steps the predator need for capture. Experiments are conducted on Intel(R) Pentium 2.20 GHz CPU with a RAM of 512M.



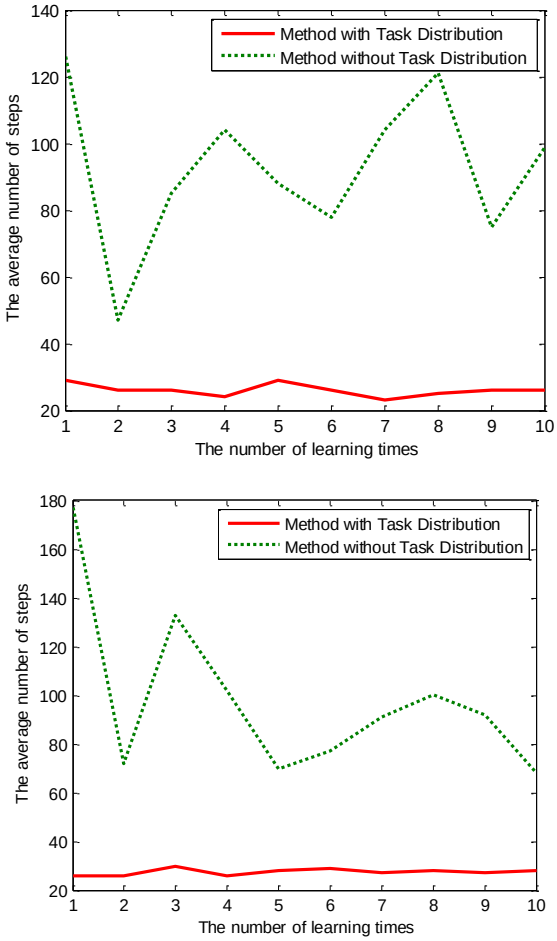




FIGURE 4. Comparison of the proposed method and traditional method

It can be seen from Figure 4 that the proposed task distribution based method has less steps for capturing the prey than the traditional method without task distribution. It is because that the proposed algorithm considers the global optimal strategy, but the traditional only pursuit the individual sub-destination which is a local strategy. So It illustrated that the effectiveness of the proposed algorithm.

## 5 Conclusions and Future Works

In this paper we presented a two-stage Q-algorithm for Multi-Agent environment in the case of predator-prey pursuit game, in this algorithm, it dynamically divides tasks for each Agent, that is to say some sub-destinations are distributed to each Agent before action selection, and then each Agent selects an action according to its sub-task. The algorithm divides the learning procedure into two parts, one is to learn tasks division strategy, and the second is to learn action selection strategy. To improve the efficiency of learning, Q values are shared by multiple Agents. Though the proposed algorithm improves the learning efficiency, the learning process of each Agent is distributive and independent, so the future work will be concentrate on making the multiple Agents more collaboratively to achieve more global optimal solution in terms of the whole multi-Agents teams.

## 6 Acknowledgments

This research is supported by the Fundamental Research Funds for the Central Universities (TD2014-02).

## References

[1] M. Mitchell. 1997 Machine Learning. McGraw-Htill. 367-84
[2] Avraham Bab, Ronen I. Brafman. 2008 Multi-Agent Reinforcement Learning in Common Interest and Fixed Sum Stochastic Games: An Experimental Study. *Journal of Machine Learning Research*, **9**(12), 2635-75.
[3] Sachiyo Arai, Katia Sycara. 2005 Effective Learning Approach for Planning and Scheduling in Multi-Agent Domain, *Proceedings of the 6th ISAB*. 507-16.
[4] Cai Q, Zhang B. 2000 An agent team based reinforcement learning model and its application . *Journal of Computer Research and Development*, **37**(9),1087-93.
[5] Wang C, Jing N, Li J, Wang J, Chen H. 2011 An algorithm of cooperative multiple satellite mission planning based on multi-agent reinforcement learning. *Journal of National University of Defense Technology*, **33**(1), 53-58.
[6] Liu C, Tan Y, Liu C, Ma Y. 2010 Application of multi agent reinforcement learning in robot soccer. *ACTA Electronica Sinica*, **38**(8), 1958-62.
[7] Tang H, Wan H, Han J, Zhou L. 2010 Coordinated look-ahead control of multiple CSPS systems by multi agent reinforcement learning . *ACTA Automatica Sinica*, **36**(2), 289-96.
[8] Xiao Z, Zhang S. 2008 Reinforcement learning model based on regret for multi agent conflict games. *Journal of Software*, **19** (11), 2957-67.
[9] Wang W, Xiao S, Meng X, Chen Y, Zhang W. 2010 Model and architecture of hierarchical reinforcement learning model based on agent , *Journal of Mechanical Engineering*, **46**(2), 76-82.
[10] Y Zhu, D Liu, G Chen, H Jia, H Yu. 2013 Mathematical modeling for active and dynamic diagnosis of crop diseases based on Bayesian networks and incremental learning. *Mathematical and Computer Modeling*, **58**(3-4), 514-23.
[11] Y Zhu, D Liu, H Jia, D. Trinugroho. 2012 Incremental Learning of Bayesian Networks based on Chaotic Dual-Population Evolution Strategies and its Application to Nanoelectronics. *Journal of Nanoelectronics and Optoelectronics*, **7**(2),113-18.
[12] Y Zhu, D Liu, H Jia, Y Huang. 2011 Structure Learning of Bayesian Network with Bee Triple-Population Evolution Strategies. *International Journal of Advancements in Computing Technology*, **3**(10), 140-48.
[13] Y Zhu, D Liu, H Jia. 2011 A New Evolutionary Computation Based Approach for Learning Bayesian Network. *Procedia Engineering*, **15**(8), 4026 - 30.
[14] Zhang S, Shi C. 2005 A multi agent reinforcement learning model based on role tracking. *Journal of Computer Research and Development*, **42**(2), 203-209.
[15] Zhou P, Hong B, Huang Q. 2006 A novel multi agent reinforcement learning approach. *ACTA Electronica Sinica*, **34**(8), 1488-91.
[16] Wang C, Yin X, Bao Y, Yao L. 2005 A shared experience tuples multi-agent cooperative reinforcement learning algorithm. *Pattern Recognition and Artificial Intelligence*, **18**(2), 234-39.

## Authors

**<Qiao Sun >, <1977.06>,< Changchun City, Jilin Province, P.R.China>**

**Current position, grades: the Lecturer of School of Information Science & Technology, Beijing Forestry University, China.**
**University studies: received her B.Sc. and M.Sc.in Electrical Engineering from Chanchun University of Technology in China. She received her PHD from Beihang University in China.**
**Scientific interest: Her research interest fields include machine learning and data mining.**
**Publications: more than 6 papers published in various journals.**
**Experience: She has teaching experience of 12 years, has completed two scientific research projects.**

**< Zhibo Chen>, <1967.01>,< Rizhao City, Shandong Province, P.R. China>**

**Current position, grades: the Professor of School of Information Science & Technology, Beijing Forestry University, China.**
**University studies: received his PHD in School of Information Science & Technology, Beijing Forestry University, in China.**
**Scientific interest: His research interest fields include internet of things.**
**Publications: more than 34 papers published in various journals.**
**Experience: He has teaching experience of 23 years, has completed 22 scientific research projects.**

**< Feixiang Chen>, <1977.11>,<Jingmen City, Hubei Province, P.R. China>**

**Current position, grades: the Professor of School of Information Science and Technology, Beijing Forestry University, China.**
**University studies: received his B.Sc. and M.Sc. in Computer Science and Technology from China University of Geosciences. He received his PhD. from Chinese academy of sciences.**
**Scientific interest: His research interest fields include Moible GIS, 3D GIS.**
**Publications: more than 30 papers published in various journals.**
**Experience: He has teaching experience of 8 years, has completed 6 scientific research projects.**

**<Fu Xu>, <1979.07>, <Weihai City, Shandong Province, P.R. China >**

**Current position, grades: the Associate Professor of School of Information and Technology, Beijing Forestry University, China**
**University studies: received her B.Sc. and M.Sc.in Electrical Engineering from Chanchun University of Technology in China. She received her PHD from Beihang University in China.**
**Scientific interest: Her research interest fields include machine learning and data mining.**
**Publications: more than 6 papers published in various journals.**
**Experience: She has teaching experience of 12 years, has completed two scientific research projects.**

**< Yanan Shi >, <1993.12>,< Linfen City, Shanxi Province, P.R. China>**

**Current position, grades: undergraduate student, College of Computer Science and Technology, Jilin University, China.**
**University studies: will receive her B.Sc. in Computer Science and Technology from Jilin University in China.**
**Scientific interest: Her research interest fields include machine learning**
**Experience: she is doing one scientific research project.**