Modeling and Analysis of Agents with Q-learning in Agent-Based Simulations

Nobuhide Nakano
Faculty of Management, Otemon Gakuin University
2-1-15 Nishi-Ai, Ibaraki, Osaka 567-8502, Japan
Tel: +81-72-641-9530, Fax: +81-72-643-9432
E-mail: nakano@res.otemon.ac.jp

Abstract

It is well known that the learning effects of agents are important on the diffusion process of rumors in social science. Though these processes can be simulated by agent-based simulations, we cannot disregard the learning effects of agents. In this paper, we deal with an agent-based problem including Q-learning, a kind of reinforcement learning. At first, we design action rules and Q values of agents. Agents decide their attitudes by attitudes which neighbor agents express, and then Q values are updated. Finally, by performing and analyzing simulation studies the effect of proposed algorithm is confirmed.

1 Introduction

Nowadays, agent-based simulations have been concentrating our attention as one of interesting and important topics in various fields, for example, social and economic activities, marketing research, and so on. Agent-based simulation is the method for analyzing various phenomena in virtual spaces which researchers set up. Some of scientists perform their researches using agent-based simulations [1]-[6].

In social science and marketing research, some researchers study diffusion processes of innovations. Rogers [7] have been extensively considered in such problems for innovation processes and communication. In conventional agent-based simulations treated diffusion process, agents decide their attitudes by their action rules, which are not changed fundamentally though action rules of agents sometimes change unexpectedly [1], [2].

But we cannot disregard the effect of learning when we consider diffusion process because agents consider and learn their actions by the rumors obtained from neighbor agents not only in a good sense but also a bad sense. That is to say, agents may change their action rules by learning.

In this paper, the effect of agent-based simulations with Q-learning is modeled and analyzed [8]. Two types of agent, pioneer and follower, are defined. Then they perform and decide their attitudes according to the performance rules which are decided by the number of neighbors [1], [2]. In this study, we introduce action rules modeled by Q value. By updating Q values after deciding their attitudes, the characters (action rules) of agents are changed by the behavior of neighbor agents with time. After some simulation studies are performed, the simulation results and analysis are shown.

2 Proposed Model

2.1 Modeling of Agents and Virtual Space

In this study, to simplify the modeling of agents two types of agents, pioneers and followers, are employed [1], [2], [6] though Rogers [7] categorizes agents at five segments. "Pioneers" are agents which feel less resistance for employing new objects, that is to say, reformists. On the other sides, "Followers" are agents which feel much resistance for employing new objects, they also called conservative people. Each agent shows an attitude either "acceptance" or "rejection" for a rumor (a new object). Some pioneers or followers are arranged in a virtual space which the researcher set and show their attitudes (either acceptance or rejection for a new object or topic) based on the attitudes of their neighbor agents. In this study, we suppose that agents only decide their attitudes based on the attitudes of their neighbor agents, which is called local influence [1], [2]. That is to say, local influence is the effect of rumors which is spread by word of mouth.

Figure 1 is illustrated the example of arranged agents in a virtual space. In this study, all of agents in the virtual space are only followers or pioneer, and the cell which places i-th row and j-th column is called (i, j) cell in this virtual space.

Figure 2 shows the decision process of an agent. An agent decide its attitude $b_i$ by the number of accepters $n_e (n_e = 0, 1, 2, 3$ or $4)$ in four neighbor agents at once on a time step $t$. Q value, $Q_e(n_i, b_i)$, is defined for $n_i$ and $b_i$ for the kind of agents $e$, either Follower ($=: F$) or Pioneer ($=: P$). Each agent decides its attitude among three patterns, that is to say, acceptance ($=: A$), rejection ($=: R$) and not changing attitude ($=: NC$). The initial action rules and initial values of Q values of followers and pioneers are given by Tables 1 and 2, respectively.
Table 1. Initial action rules and Q values for followers

<table>
<thead>
<tr>
<th>Rule</th>
<th>Initial values of Q values</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 R</td>
<td>$Q_e(0, A) = 0$ $Q_e(0, NC) = 0$ $Q_e(0, R) = 1$</td>
</tr>
<tr>
<td>1 R</td>
<td>$Q_e(1, A) = 1$ $Q_e(1, NC) = 0$ $Q_e(1, R) = 0$</td>
</tr>
<tr>
<td>2 NC</td>
<td>$Q_e(2, A) = 0$ $Q_e(2, NC) = 1$ $Q_e(2, R) = 0$</td>
</tr>
<tr>
<td>3 A</td>
<td>$Q_e(3, A) = 1$ $Q_e(3, NC) = 0$ $Q_e(3, R) = 0$</td>
</tr>
<tr>
<td>4 A</td>
<td>$Q_e(4, A) = 1$ $Q_e(4, NC) = 0$ $Q_e(4, R) = 0$</td>
</tr>
</tbody>
</table>

Table 2. Initial action rules and Q values for pioneers

<table>
<thead>
<tr>
<th>Rule</th>
<th>Initial values of Q values</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 R</td>
<td>$Q_p(0, A) = 0$ $Q_p(0, NC) = 0$ $Q_p(0, R) = 1$</td>
</tr>
<tr>
<td>1 A</td>
<td>$Q_p(1, A) = 1$ $Q_p(1, NC) = 0$ $Q_p(1, R) = 0$</td>
</tr>
<tr>
<td>2 NC</td>
<td>$Q_p(2, A) = 0$ $Q_p(2, NC) = 1$ $Q_p(2, R) = 0$</td>
</tr>
<tr>
<td>3 R</td>
<td>$Q_p(3, A) = 1$ $Q_p(3, NC) = 0$ $Q_p(3, R) = 1$</td>
</tr>
<tr>
<td>4 A</td>
<td>$Q_p(4, A) = 1$ $Q_p(4, NC) = 0$ $Q_p(4, R) = 0$</td>
</tr>
</tbody>
</table>

2.2 Decision Process and Update of Q value

We explain a decision process and an update of Q values by using Figure 2 [8]. We suppose that the center agent (depicted by solid line) is a follower and a rejecter in the case of Figure 2. At first, the number of accepters in neighbor four agents depicted by broken line is counted. Black circles are agents which do not concerned with the decision of the center agent. In this figure, as there are three accepters around the center agent, the agent check Q values at $n_t = 3$. In this case, we suppose he has following values:

$Q_e(3, R) = 0.1$, $Q_e(3, NC) = 0.2$ and $Q_e(3, A) = 0.8$.

In this case, $Q_e(3, A)$ is the largest of the three Q values. As we select ε-greedy method ($ε = 0.1$ in this study) for choosing action rule, The probability of selecting 'Acceptance' is nine in ten, and the probability of selecting which of 'Acceptance,' 'Not Changing (attitude)' or 'Rejection' at random is one in ten.

Updates of $Q_e(n_t, b_t)$ are performed by following algorithm [8] after deciding action rules at time step $t$:

$$Q_e(n_t, b_t) \leftarrow Q_e(n_t, b_t) + \alpha \left[ r + \gamma \max_{b_{t+1}} Q_e(n_{t+1}, b_{t+1}) - Q_e(n_t, b_t) \right],$$

where $Q_e(n_{t+1}, b_{t+1})$ shows the future Q value for the action $b_{t+1}$ and the state $n_{t+1}$. $r$ is the reward for transition from $n_t$ to $n_{t+1}$. $B_{t+1}$ is the set of whole actions at $n_{t+1}$. $\alpha (0 < \alpha \leq 1)$ is the learning rate and $\gamma (0 \leq \gamma \leq 1)$ is the discount rate.

In this study, the reward $r$ is given when all agents become accepters ($n_i = 100$) because the purpose of this study is the spread of rumors for all agents in the virtual space.

3 Simulation Method

Recalling the mentioned above, the procedure for the simulation is performed by the following steps:

[1] One of initial conditions for simulations shown in Table 3 is selected and set up.

[2] 100 agents (100 pioneers or 100 followers) are placed in a grid which has 10 rows and 10 columns and each agent is given the initial attitude (acceptance or rejection). In this study, initial accepters are arranged at central squared area in the virtual space, and other agents are all rejecters, for example, shown in Figure 1.

[3] Initial values of Q values at followers and pioneers are defined in Tables 1 and 2, respectively.

[4] Each agent decides own attitude once every one time step based on $Q_e(n_t, b_t)$ shown in Figure 2 and then updates $Q_e(n_t, b_t)$ by Equation (1).

[5] Simulations are performed until all agents become accepters, and then the reward $r$ is given at this time. If rejecters remain until the time step $t = 1000$, simulations are ended by force.

[6] Simulation processes [4] and [5] are repeated at 1000 times, and the number of finishing times at $t < 1000$ in 1000 simulations is counted at once. We define a trial as simulations performed at 1000 times.

[7] For one initial condition (shown in Table 3), 10 trails are performed. Then, the average of finishing times at 10 trials ($= A_{avg}$) is calculated for one initial condition.
4 Results and Analysis

4.1 Effects of Q-learning

Results of simulation studies \( A_{\text{ave}} \) are shown in Table 3. In these simulations, we employ \( \epsilon \)-greedy method for selecting attitudes and set \( \epsilon = 0.1 \) except simulation No. 3, which are performed only 1 trial because of performed by greedy method \( (\epsilon = 0 \text{ at } \epsilon \text{-greedy method}) \). And we set \( \alpha = 0.1 \), \( \gamma = 0.9 \) and \( \tau = 10 \).

We have found that Q-learning with \( \epsilon \)-greedy method is effective in agent-based simulations (Simulations No. 2 and 3 with simulation No. 1), and that finishing circumstances of pioneers are different from that of followers by the results of simulations No. 1 and No. 4. Furthermore, from the results of simulations No. 1, 5, 6, 7, we have found proposed algorithm works very well for various initial conditions.

From these results, we can prove that agents should learn and change their characters so as to spread rumors by performing these simulations. Furthermore, we think that Q-learning with \( \epsilon \)-greedy method is very humane algorithm because accidental events which some men happen are considered by including random parameter \( \epsilon \).

4.2 Variations of Accepters

Here, we analyze the variations of accepters according to the passage of time. Figure 3 is depicted variations of accepters at Simulations No. 1. Variations of accepters at Simulations No. 7 and 2 are illustrated in Figure 4 and 5, respectively. In these Figures, the results of 100 simulations are depicted. Then, the number of accepters becomes 0 at \( t > t_{\text{end}} \) when the number of accepters once becomes 100 at \( t = t_{\text{end}} \) and the simulation finishes by force.

At simulations No. 1, shown in Figure 3, 10 simulations are not finished. 90 simulations are finished at 354 \( \leq t \leq 500 \). The variations have some regularity at these simulations, which accepters once decrease at \( 0 \leq t \leq 200 \) and then accepters increase gradually. At simulations No. 7, shown in Figure 4, 12 simulations are not finished. 88 simulations are finished at 374 \( \leq t \leq 537 \). In this case, the regularity of the variation diminishes somewhat. The simulations that do not finish at \( t < 1000 \) show random variations. At simulations No. 2, shown in Figure 5, all simulations are not finished. In this case, the variations are all random, but accepters vary under 20 at \( t \geq 200 \).

4.3 Variations of Q values

Finally, the variations of Q values are analyzed. In this paper, Q values of the agent at (5,5) cell in Simulations No. 1 are treated, shown in Figure 1. Variations of \( Q_p(0,A) \), \( Q_p(0,NC) \) and \( Q_p(0,R) \) are illustrated in Figure 6, 7 and 8, respectively. Similarly, variations of Q values \( Q_p(n_1,A) \), \( Q_p(n_1,NC) \) and \( Q_p(n_1,R) \) at \( n_1 = 1, 2, 3 \) and 4 are illustrated in Figures 9 – 20 in regular order. In these Figures, the results of 100 simulations are depicted, respectively. Then, the Q values also become 0 at \( t > t_{\text{end}} \).
when the number of accepters becomes 100 at \( t = t_{\text{end}} \) and the simulations finish by force.

Q values at \( n_t = 0 \), 1 and 2 show similar variations. Q values of which initial values are 1, \( Q_f(0, R) \), \( Q_f(1, R) \) and \( Q_f(2, NC) \), decrease and the other Q values at \( n_t = 0 \), 1 and 2 increase a little but vary under 0.4.

Q values at \( n_t = 3 \) and 4 also show similar variation. Q values of which initial values are 1, \( Q_f(3, A) \) and \( Q_f(4, A) \) decrease, the other Q values at \( n_t = 3 \) and 4 increase a little but vary under 0.4. In Figures 15, 16, 18 and 19, variation of pulse-like shape are occurred because simulations are finished and Q values increase immediately by getting reward \( r \).

5 Conclusions

In this paper, we confirm that learning effects of agents can be expressed by the proposed algorithm. The point is that the agents with Q values can frequently change their characters by updating Q values.

We will be able to design the degree of variations of characters by adjusting various parameters, learning rate, discount rate, and so on. Though introduction of reinforcement learning is very important, the method of action select also becomes the key point of these simulations because stochastic process can be applied in this point.

References

Figure 9. Variations of $Q_F(1, A)$ at Simulation No. 1

Figure 10. Variations of $Q_F(1, NC)$ at Simulation No. 1

Figure 11. Variations of $Q_F(1, R)$ at Simulation No. 1

Figure 12. Variations of $Q_F(2, A)$ at Simulation No. 1

Figure 13. Variations of $Q_F(2, NC)$ at Simulation No. 1

Figure 14. Variations of $Q_F(2, R)$ at Simulation No. 1
Figure 15. Variations of $Q_F(3, A)$ at Simulation No. 1

Figure 16. Variations of $Q_F(3, NC)$ at Simulation No. 1

Figure 17. Variations of $Q_F(3, R)$ at Simulation No. 1

Figure 18. Variations of $Q_F(4, A)$ at Simulation No. 1

Figure 19. Variations of $Q_F(4, NC)$ at Simulation No. 1

Figure 20. Variations of $Q_F(4, R)$ at Simulation No. 1