Clustering Agent Based on Preference Measure for Reputation Systems

Yizhi Ren†, ‡ Mingchu Li† Yongrui Cui† Kouichi Sakurai‡

†School of Software, Dalian University of Technology
Dalian, 116621, P.R.China

‡Department of Informatics, Kyushu University
Hakozaki, Fukuoka 812-81, Japan
ren@itslab.csce.kyushu-u.ac.jp
sakurai@inf.kyushu-u.ac.jp

Abstract The formation of agent group is one of the important research topics in agent-based applications. Traditional researches can hardly be adaptable to each separate agent's needs and preferences which are common concerns in social networks. This paper proposes a method to group agents based on the preference measure for reputation system. These results are applied to reflect the difference of agents' preference and improve the sensitiveness of detecting agents' behavior.

1 Introduction

1.1 Background and Motivation

In many agent-based applications (e.g. reputation systems), agents have identical functionality and play the role in both servers and clients. When an agent wants to trade with other agents, it will evaluate the target agents' credit and make trust decisions according to its direct experience or other agents' feedback. A trust computation model, which responsible of searching the trust related information, making trust policies, and so on, is one of the key components in the reputation systems. Thus, how to building the trust model has become one of the hot research topics in recent years. Existing works are very different according their various methods of trust description, evaluation, reasoning, and so on. Although recent research has made some progress both in theory and application fields, most of them evaluate agents behaviors very roughly and without consider agents' various preferences and specific applications. Some basic questions needed to be considered before dealing with this problem. Such as: the effect of agents' preference to their behaviors; the way to collect the agents' preference; the method to measure the similarity of preference among agents, and so on.

1.2 Related Works

The concept of similarity is fundamentally important in almost every scientific field. If we can measure similarity, people can differentiate objects, group the objects into clusters, classify a new object into group and predict the behavior of new object, etc. Generally, we can classify the approaches into three main categories[1]: **Goal-based methods** (e.g., [2]) appear suitable when it's necessary to form groups o agents that cooperate for achieving a common goal, and the similarity measure in this context are based on comparisons of the different agents' goal. **Homogeneity of idea based methods**(e.g., [3]) are used to form social communities of agents that desire to share resources and ideas about common subjects. **Knowledge representation based methods** (e.g., [1] [4]) group the agents who should be similar with point of view of the knowledge representation.

1.3 Challenging Issues

Despite the strong sociological foundation for the concepts of trust and reputation, existing computational models for them are often not grounded on understood social characteristics of these quantities. Some key challenge issues about the agent preference in the reputation systems are needed to be resolved as below: (1) How to obtain and select suitable invariant or descriptors to characterize trust relationship to compare trust difference effectively. (2) The calculations of some effective invariant or descriptors become more and more difficult with the lengths of the matrices longer. (3) How to cope with multivariate trust evaluation in reality?
And so on.

1.4 Our Contributions

Preference similarity measure may provide a good solution to understand agents' behaviors and help to design an efficient mechanism to enhancing the cooperation among agents. In this paper, we study the problem of defining similarity measure on preference in reputation systems, and then discuss an applications of clustering agent' preference in the reputation systems. The main contributions in this work are list as follows: (1) Considering the multivariate data when evaluate the service satisfaction, we propose a method to detect the agents' preference using the normalized harming distance as the basis for comparison. (2) Providing a preference-based agents' classification method based on the defined coefficient (e.g. as preference similarity and preference relevance).

1.5 Comparing Our New Results to Related Works

Our method can provide local and global similarity measure to accommodate the various agents' preferences in the reputation systems, and also provide a flexible way to present differentiated agents and combine different aspects of service requirements.

The remainder of the paper is organized as follows: In section 2, we present an agent preference detecting algorithm using fine-grained service; in section 3, a preference similarity measure method will be proposed; in section 4, we will give an application example using our method. Finally, we will make conclusions of our work.

2. Preference Detecting Using Fine-grained Service

Let *Service* be *n*-dimensional vector $sev(x_1, x_2 ..., x_n)$, where x_k ($k \in [1,n]$) is value the k^{th} property. [5] introduce a Gauss-bar function to evaluate the similarity of service satisfaction. However, in real systems the value of x_k may not be quantitative data, but nominal data type, ordinal scale, or binary scale, and so on. How we can aggregate mixed type of data (multivariate data) up to *n* dimension? We will give a method to detect agent's preference using fine-grained service satisfaction in this section.

2.1 Normalized Multivariate Data

Supposed that agent *A* wants to discovery whether other agent with the preferences about service type α_i or not. Before measuring that, *A* should set the standard sev_{α_i} which service value about service type α_i in advance.

According the data type property x_k ($k \in [1,n]$) in $sev_i(x_1, x_2 ..., x_n)$, there are several cases as following:

Quantitative data type. If x_k is quantitative data type, we don't need to change anything but normalize it. For service set $\{sev_1, sev_2, ..., ser_m\}$, we get the $m \times m$ normalized matrix $M^1 = (m^1{}_{ij})$, $i, j \in [1,m]$ such that (see Eq. 1):

$$m^1{}_{ij} = \frac{|sev_i(x_k) - sev_j(x_k)|}{\max\{|sev_{i'}(x_k) - sev_{j'}(x_k)| : i', j' \in [1,m]\}} \qquad (1)$$

Where $sev_i(x_k)$ and $sev_j(x_k)$ are *k*-th property value of sev_i and sev_j, respectively; $\max\{|sev_{i'}(x_k) - sev_{j'}(x_k)| : i', j' \in [1,m]\}$ denotes the maximum value of difference of *k*-th property value in service set $\{sev_1, sev_2, ..., ser_m\}$.

Ordinal data type. Supposed that x_i is ordinal data type, we get the rank of these data and normalize the rank into range [0, 1]. E.g. Given the set $\{sev_1(x_i), sev_2(x_i), ..., ser_m(x_i)\}$, we order the set and change it into new set $\{sev_1'(x_i), sev_2'(x_i), ..., ser_m'(x_i)\}$. Finally, we can map the new set into a ranked set $\{1, 2, ..., k\}$, $k \in [1,m]$.

We have the following proposition:

Proposition 1:

(1) If $k < m$, $\exists sev_{i'}(x_i) = sev_{j'}(x_i)$ ($i' \neq j'$ and $i', j' \in [1,m]$), then $sev_{i'}$ and $sev_{j'}$ have the same rank value.

(2) If $k = m$, then elements in set $\{sev_1'(x_i), sev_2'(x_i), ..., ser_n'(x_i)\}$ all are different.

Supposed that $sev_j(x_i)$'s rank value is *t*, then the normalized rank value of $sev_j(x_i)$ is (see Eq. 2):

$$\text{Nor\_value}(sev_j(x_i)) = \frac{t-1}{r-1} \qquad (2)$$

The computing of normalized matrix $M^2 = (m^2{}_{ij})$ ($i, j \in [1,m]$) of ordinal data is the same as the Eq.1.

Nominal data type. If x_i belongs to the nominal data type, and there are two cases:

Case 1: The nominal data is mutually exclusive values. Because they are mutually exclusive, it would be better if we assign each value of category into several binary dummy variables. The number of dummy variable can be calculated by $\lceil \log(num\_choice) \rceil$, where *num_choice* denotes the number of nominal data choice.

Because x_k is mutually exclusive nominal data type, each $sev_j(x_k)$ ($j \in [1,m]$) can be denoted by $\{0,1\}^t$, then we get the $m \times m$ normalized matrix $M^3 = (m^3_{ij})$, $i, j \in [1, m]$ such that(see Eq. 3):

$$m^3_{ij} = \frac{ham\_dis(ser_i(x_k), sev_j(x_k))}{\lceil \log(num\_choice) \rceil} \qquad (3)$$

Where $ham\_dis(ser_i(x_k), sev_j(x_k))$ denotes the hamming distance between vector $ser_i(x_k)$ and $sev_j(x_k)$.

Case 2: The nominal data is not mutually exclusive values, that is to say the choices can be multiple choices, and an agent may have several activities in the target agent's resource.

Supposed that the number of choices be *t*, then $sev_j(x_k)$ ($j \in [1, m]$) can be denoted by $\{0,1\}^t$. We get the $m \times m$ normalized matrix $M^4 = (m^4_{ij})$, $i, j \in [1, m]$ such that (see Eq. 4):

$$m^4_{ij} = \frac{ham\_dis(sev_i(x_k), sev_j(x_k))}{t} \qquad (4)$$

Where $ham\_dis(sev_i(x_k), sev_j(x_k))$ denotes the hamming distance between vector $ser_i(x_k)$ and $sev_j(x_k)$.

2.2 Aggregating the Normalized Matrix

Actually, each property may have different weight value according to the different agents. Let $W(w_1, w_2, ..., w_n)$ be weight vector of $sev(x_1, x_2, ..., x_n)$, where $\sum_{i=1}^{n} w_i = 1$. So the aggregation matrix $M = (m_{ij})$, $i, j \in [1, m]$ such that (see Eq. 5):

$$m_{ij} = \sum_{k=1}^{n} m^k{}_{ij} w_k \qquad (5)$$

2.3 Analysis the Agent Preference

Assuming that the *i*-th row in aggregation matrix *M* is the result of A's. For each *j*($j \in [1, m]$), if $m_{ij} < \gamma$ ($i \neq j$), then agent A think that the service ser_j provider with the preference about the service type α_i. The same service provider may be with preference conflict happened in the same evaluation process. If this case happens, A can not judge the provider's preference this time, but record it for further analysis next time.

Repeated the above evaluation process, agent A can get other agents' preference about certain service type by statistical analysis.

3. Preference Similarity Measure

According to the above section, we get the preference from the service similarity measure. Then, how can an agent use these data to evaluate other agents' preference from its own point of view? In this section, we will propose a preference similarity measure method.

3.1 Preliminaries

Definition 2: *Preference*, *p*, is binary relations which the choice between alternatives based on satisfaction, enjoyment, utility and so on.

We define the preference in fine-grained style, which can be represented as *n*-dimensions vector:

$$p = (\alpha_1, \alpha_2, .., \alpha_n)$$

Here α_i denotes *i*-th service type and $\alpha_i \in \{0,1\}$. If $p(\alpha_i) = 1$, we say that the agent with the α_i preference, otherwise without the α_i preference.

In order to classify the agents with different preferences, we introduce the concept of *Cluster*, which is defined as follows:

Definition 3: *Cluster* is group of agents toward an approximate similarity preference.

According the lifecycle of *Cluster*, we can divide it into three types:

1) *Temporary cluster* denotes the agent groups whose lifecycle within a task.
2) *Median cluster* denotes the agent groups whose lifecycle lasting for several tasks.
3) *Long-term cluster* denotes the most trusted agents or recognized honest agents who have the long lifecycle.

Agents in the different type of clusters are not static, and there may be with dynamic transition among them under certain conditions. For example, supposed that the preference classification mechanism of agent *e* added the agent *u* into its p_e preference cluster, which is a temporary cluster. If the agent *u* is trustworthy in actual transaction (*e* can get the information from its own experience or recommendation from others in the network), then *e* will add *u* into its median cluster about $p_e \in P_e$ preference; If agent u is still trusted after several times transaction (based on agent *e*'s need), then it will be added in to *e*'s long-term cluster about $p_e \in P_e$ preference.

On the contrary, if the agent u is dishonest in the trade, e may add it into Blacklist, which records the malicious agents. The details of transition process can see Fig.1:

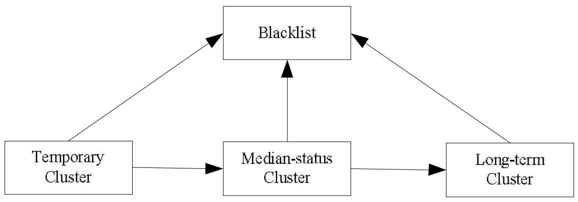

Figure 1.Cluster transition

3.2 Preference Similarity Measure

In the real network, the capability of agents can be various because of their different processing capacity, storage space, bandwidth, congestion condition and so on. We divide these agents into two parts: *Rationality agent* and *Bounded Rationality Agent*.

Rationality Agent is powerful agents' set where each agent can know other agents preference, and can get somehow optimally in pursuit of its goals.

It's very easy for an agent belonged to *Rationality Agent* to find other agents with the approximate preference.For example: let $p_u = (\alpha_1, \alpha_2, .., \alpha_n)$ be the preference of agent u. For i=1…n, if $\left(p_u(x_i) = p_e(x_i)\right) = true$, then add u into p_e cluster; otherwise delete u.

Our discussion thus far has dealt with similarity of preferences in the case of certainty. However, the most potential applications require the ability to define similarity of preferences under uncertainty, E.g. reputation system. In order to accommodate these problems, we assume that most agents in the network are bounded rationality.

Bounded Rationality Agent is bounded agents set where each agent with incomplete knowledge, limited power of computation, and so on.

So, agents belonged to *Bounded Rationality Agent* may only know part of target agent's preference.

Let $U = \{u_1, u_2, ..., u_t\}$ be set of agents in the network, and p_{u_i} be the preference of agent u_i in U. So, there exists some uncertain factor of the target agent's preference. For example, the preference of u_i may be described as $p_{u_i} = (1, -, 0, -, ..., -)$ from the angle of agent e, where the symbol '-' denotes the unknown part of preference about that kind of service type, '1' with that service type preference, and '0' without that service type preference.

In order to deal with these uncertain factors, we define the following concepts:

- R is the number of common preferences shared in both p_{u_i} and p_e, that means the number of $p_{u_i}(\alpha_k) = p_e(\alpha_k) = 1$, where k=1, 2, …, n.
- S is the number of common preferences shared in both p_{u_i} and $\overline{p_e}$, that means the number of $p_{u_i}(\alpha_k) = \overline{p_e}(\alpha_k) = 1$, where k=1, 2, …, n.
- T is the number of common preferences shared in both $\overline{p_{u_i}}$ and p_e, that means the number of $\overline{p_{u_i}}(\alpha_k) = p_e(\alpha_k) = 1$, where k=1,2,...,n.

Here $\overline{p_e}$ and $\overline{p_{u_i}}$ are the complement of p_e and p_{u_i}, respectively. We assume that the complement of uncertain part of property is unchanged. E.g: if $p = (1, -, 0, 1)$, then its complement is $\overline{p} = (0, -, 1, 0)$.

We have the following proposition:

Proposition 2: $T + S \in [0, n]$, and if $T + S = 0$ means that vector $p_{u_i} = p_e$.

Proof. The preference vector is n-dimensional, and we can easily conclude that $0 \le T + S \le n$.

Another, since $T + S = 0$, we have T=0 and S=0. If $p_{u_i}(\alpha_k) = 0$ and T=0, then $p_e(\alpha_k) = 0$. Or else, if $p_{u_i}(\alpha_k) = 1$ and S=0, then $p_e(\alpha_k) = 1$. So for each $k \in [1, n]$, we have $p_{u_i}(\alpha_k) = p_e(\alpha_k)$, then $p_{u_i} = p_e$.

Definition 4: *Preference similarity coefficient, PSC,* denotes the preference similarity between the target agent and the source agent.

The *PSC* of between p_{u_i} and p_e can be computed as bellows (see Eq. 6):

$$PSC(p_{u_i}, p_e) = \frac{R}{R+S+T} \qquad (6)$$

Definition 5: *Preference relativity coefficient, PRC,* denotes the preference relativity between the source agent and target agent.

The *PRC* between p_{u_i} and p_e can be computed as follows (see Eq. 7):

$$PRC(p_{u_i}, p_e) = \frac{R}{R+S} \qquad (7)$$

The *PRC* was used to divide the bigger cluster into several sub-clusters which with properties of increasing similarity to the agent's preference.

Definition 6: For agent set U ($|U| = m$) and agents' preference set P ($|P| = t$), the $m \times t$ comparison matrix $CM = (c_{ij})$ between them is given as bellows:

$$c_{ij} = \begin{cases} 1 & \text{if PPS}>\partial_1 \text{ and PSC}>\partial_2 \\ 0 & \text{otherwise} \end{cases}$$

Here ∂_1, ∂_2 is a threshold value of preference similarity and relativity, respectively.

After getting the comparison matrix, we can perform clustering on its. The details of how to cluster these data according the above results are not the main topics in our work, and more can refer [5].Finally, we can get corresponding clusters of different preference cluster.

4 An Application Example

In the previous sections, we have given the preference detecting and classifying method. Here we examine an application of these results to the trust decisions in reputation systems.

Before the example, we give the framework of agent preference clustering using our method, see Fig.2.

Let each service be 4-dimensonal vector $serv(cnd_1, cnd_2, cnd_3, cnd_4)$, here cnd_i ($i \in [1,4]$) denotes the concrete property of service. In our example, we assume service to be:

$$serv(time, speed\_mode, activity, congestion)$$

Here:

1. *time* is a quantitative data type which measured in minutes, and it records the length of trade time required;
2. *speed_mode* is nominal data type, and it denotes the bandwidth that the service provided. Assuming that there consists of four choices of speed mode: {Very high bandwidth(>100M), High bandwidth ([10M, 100M]), Ordinary speed ([1M, 10M]), Low speed (<1M)} .The above four choices are mutually exclusive, that is only one speed mode for once service;
3. *activity* is nominal data type, and it records the activity in consist of 6 choices of activity: {download, Access privacy data, running program, modifying data, deleting data, reading data}.The choices are multiple choices, that one agent may have several activities in the target agent's resource;
4. *congestion* is ordinal scale with 5 values:{-2, -1, 0, 1, 2}, here '-2' , '-1', '0', '1' and '2' mean very dissatisfied, dissatisfied, indifferent, satisfied and very satisfied, respectively. It measures agents satisfaction toward the target services' network status;

Supposed that there are four services {*serv_A*, *serv_B*, *serv_C*, *serv_D*} in the network, and the initial service vectors are given in the Table 1.

Table 1: Initial services vector

	Time	Speed mode	Activity	Congestion
Serv_A	**20**	**2**	**1,2,4**	**1**
Serv_B	50	1	6	2
Serv_C	10	4	3,4	-1
Serv_D	60	4	2,5	0

As see in Table 1, there are multivariate data types in the service vector. So, we should transform this data into coordinate style. The details are list as follows:

Time: we have the comparison matrix about *time* ($CM_{time} = (m_{ij})$) among A, B, C and D as follows:

$$CM_{time} = \begin{pmatrix} 0 & 30 & 10 & 40 \\ 30 & 0 & 40 & 10 \\ 10 & 40 & 0 & 50 \\ 40 & 10 & 50 & 0 \end{pmatrix}$$

So, we can get the normalized matrix NM_{time} according the formula (1) as follows:

$$NM_{time} = \begin{pmatrix} 0 & 0.6 & 0.2 & 0.8 \\ 0.6 & 0 & 0.8 & 0.2 \\ 0.2 & 0.8 & 0 & 1 \\ 0.8 & 0.5 & 1 & 0 \end{pmatrix}$$

Congestion: Mapping the choice of congestion from (-2, -1, 0, 1, 2) to (0, 1, 2, 3, 4) by each element in above adding 2. Then, the normalized rank of each service can be (0, 1/4, 1/2, 3/4, 1) according the formula (2). So, the *congestion* of A, B, C and D is 3/4, 1, 1/4 and 1/2 respectively. So, we get the congestion normalized matrix as follows:

$$NM_{congestion} = \begin{pmatrix} 0 & 0.25 & 0.5 & 0.25 \\ 0.25 & 0 & 0.75 & 0.5 \\ 0.5 & 0.75 & 0 & 0.25 \\ 0.25 & 0.5 & 0.25 & 0 \end{pmatrix}$$

Speed Mode: it belongs to the mutually exclusive nominal data type. Based on the following formula:

$$m = \lceil \log(num\_choice) \rceil$$

Here *num_choice* denotes the number of nominal data choice. So, we have $m = \lceil \log 4 \rceil = 2$. and *speed_mode* normalized matrix can be denoted as below:

$$NM_{speed\_mode} = \begin{pmatrix} 0 & 0.5 & 0.5 & 0.5 \\ 0.5 & 0 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0 & 0 \\ 0.5 & 0.5 & 0 & 0 \end{pmatrix}$$

Activity: We use *n* (the number of choice) dimensional vector in binary scale. E.g. *service* A's *activity* can be denoted as: (1, 1, 0, 1, 0, 0). According the formula (4), we can get the *activity normalized matrix* as follows:

$$NM_{activity} = \begin{pmatrix} 0 & 0.67 & 0.5 & 0.5 \\ 0.67 & 0 & 0.5 & 0.5 \\ 0.5 & 0.5 & 0 & 0.67 \\ 0.5 & 0.5 & 0.67 & 0 \end{pmatrix}$$

Supposed that agent *A* sets the weighted value of properties in service $serv(time, speed\_mode, activity, congestion)$ is $(0.2, 0.3, 0.1, 0.4)$, and then we can get the aggregation matrix by formula (5):

$$AM = \begin{pmatrix} 0 & 0.44 & 0.44 & 0.49 \\ 0.44 & 0 & 0.64 & 0.44 \\ 0.44 & 0.64 & 0 & 0.55 \\ 0.49 & 0.44 & 0.55 & 0 \end{pmatrix}$$

If the tolerant threshold of *Serv_A* is 0.48, then agent A thinks that B and C with the preference about the serve type α_i, while D without.

Repeated the above process, and record the result by statistical analysis. Supposed that agent A's preference is $p_A = (1, 0, 1, 0, 0, 1, 1)$, and the target agent preference is $p_t = (1, 1, 0, -, 0, 1, 1)$, here '-'denotes uncertain part. Thus, $\overline{p_t} = (0, 0, 1, -, 1, 0, 0)$ and $\overline{p_A} = (0, 1, 0, 1, 1, 0, 0)$. According the definition in the section 3.2, we have:

$$\begin{cases} R=4 \\ S=2 \\ T=2 \end{cases}$$

Also,

$$PSC(p_A, p_t) = \frac{R}{R+S+T} = 0.5$$

$$PRC(p_A, p_t) = \frac{R}{R+S} \approx 0.67$$

Assuming that $\partial_1 = 0.4$ and $\partial_2 = 0.5$, then we have:

$$\begin{cases} PSC > \partial_1 \\ PRC > \partial_2 \end{cases}$$

Thus, the target agent has the same preference with agent A in a sense.

5 Conclusions

We propose a method of preference similarity measure in trust model, which develop to reflect the difference of agents' preference and improve the sensitiveness of detecting agents' behaviors.

Acknowledgement

The first author of this research is supported by the governmental scholarship from China Scholarship Council, Nature Science Foundation of China under grant No.: 60673046 and 90715037, University Doctor Subject Fund of Education ministry of China under grant No.: 200801410028, and Natural Science Foundation Project of Chongqing, CSTC under grant No.: 2007BA2024.

References

[1] Salvatore Garruzzo and Domenico Rosaci: Agent clustering Based on Semantic Negotiation. ACM Transactions on Autonomous and Adaptive Systems, Vol. 3, No.2, Article 7, May 2008.

[2] Vassileva, J. Breban, S. and Horsh, M. Agent reasoning mechanism for long-term coalitions based on decision making and trust. Comput. Intell. 18, 4, 583-595, 2002.

[3] Villail, M.B., Musse, S. R., and De Oliveira, L. P.L. A model for generating groups of virtual agents. In proceedings of the intelligent Virtual Agents (IVA 2003). 164-169, 2003.

[4] Rosaci, D. An ontology-based two-level clustering for supporting e-commerce agents activities. In Proceedings of E-commerce and Web Technologies, (EC-web 2005), vol 3590, 31-40.2005.

[5] Onwubolu, G.: Redesigning job shops to cellular manufacturing systems. Integrated Manufacturing Systems 9(6), 377-382, 1998