Available online at www.isas.org.in/jisas
JOURNAL OF THE INDIAN SOCIETY OF AGRICULTURAL STATISTICS 67(1) 2013 23-32

A Reputation based Service Provider Selection System for Delegation of Job by Farmers

Punam Bedi, Harmeet Kaur and Bhavna Gupta
University of Delhi, Delhi

Received 30 September 2011; Revised 20 October 2011; Accepted 2 February 2012

SUMMARY

Selection of a service provider, to perform agriculture jobs, is a major challenge for any farmer. This problem of selection of a service provider to delegate the job is being addressed by a balanced reputation based service provider selection system for farmer. Being agent based system, agents compute the reputation of the service providers present in the e-community based on their past experiences and recommendations collected from their trustworthy acquaintances. Further it is observed that if selection is done only on the basis of the reputation of the service providers then this may introduce delay in the accomplishment of the job because of overloaded reputed service providers. This paper presents a scheme to distribute the work to reputed service providers in such a way that delay in the accomplishment of the job can be minimized. As reputation is a subjective term, so to quantify reputation the concept of Intuitionistic Fuzzy Sets (IFS) is used in this paper. Further the Intuitionistic Fuzzy distances among the recommendations of various trustworthy peers are computed using which the trust on a trustworthy acquaintance is updated.

Keywords: Trust, Reputation, Agent, Delegation of job, Intuitionistic Fuzzy Set.

1. INTRODUCTION

In e-agriculture farmers can delegate some of their tasks like pest/weed control, watering of plants and cultivating seasonal crops to available service providers. But due to the availability of many choices in electronic communities, it becomes difficult to decide to whom job should be delegated. Traditionally prior to delegation of the job due to the interaction between the parties, a trust is established between them which helps in selection. Thus trust established between parties is the key factor for selection of the vendor. Similarly, problem of selection of service provider in electronic communities also get solved if trustworthiness of various available service providers can be computed. Considering only trust factor while delegating the job may introduce delay in accomplishment of the job because reputed service providers may already be overloaded with the jobs assigned to them. In this paper, a personalized balanced agent based reputation system for farmers to delegate the job is proposed in which every farmer is assisted by an agent known as farmer agent. In our system, agent stores the personal experience and also maintains a list of its trustworthy acquaintances. On getting a request from the farmer to delegate a job its corresponding farmer agent on the basis of requirement specification prepares a query in the form of request vector. This vector is passed to its trustworthy agents to get recommendations. These agents may also send this vector further to their trustworthy agents and so on keeping in consideration a *timeout* variable. The amount of the time any farmer can wait to have the result of the request is stored in timeout variable. Its initial value is specified by the

Corresponding author : Punam Bedi
E-mail address : pbedi@cs.du.ac.in

farmer himself at the time of the requirements specifications.

Every agent that gets the request vector then computes the reputation of those service providers who are capable of satisfying that request and with whom it has interacted directly and has found them to be satisfactory. This computed reputation and capability of service providers is sent as recommendation to the requesting agent after aggregating the recommendations from its trustworthy acquaintances. In this process of aggregation of recommendation not only the service providers with whom direct interaction of the agent has taken place are considered but some of those service providers with whom the agent has not interacted before, also get considered. This is similar to what happens in our society. For selecting some services we not only use our personal experience but we consider that information also which we have acquired unintentionally while interacting with others. Therefore, we assume that in idle time agents also interact with other agents and acquire information about service providers from them. Although a lot of fuzziness and uncertainty is involved in this information but it cannot be ignored altogether, what if that information is correct.

Once the aggregated recommendations from the trustworthy acquaintances, about the capability and reputation of the various service providers, are given to the farmer agent, lists are prepared based on capability and reputation of the service providers. Finally, the aggregation of the lists is done based on the requirement specified by the farmer for job delegation. This process helps in reducing the delay in accomplishment of the job which may get introduced if the job is distributed only on the basis of the reputation of the service providers. Because it may happen that reputed service providers may already be overloaded with the jobs assigned to them and they are buffering the other jobs as in real life reputed shopkeepers are normally busy and take time in completion of the task.

For computing reputation of various service providers and to handle the fuzziness and uncertainty in the information the concept of Intuitionistic Fuzzy Sets (IFS) is used.Intuitionistic Fuzzy Sets (IFS) are one of the interesting and useful generalizations of fuzzy set theory, introduced by Atanassov (1999) having membership, non membership and hesitation part. Fuzzy sets are IFS but the converse is not true. In fact there are situations such as the problem of reputation computation where IFS theory is more appropriate to deal with because of the hesitation part present in it (Bedi and Kaur 2004).

Rest of the paper is organized as follows. Basics of Intuitionistic Fuzzy Sets (IFS) are in section 2. Section 3 presents the related work. Section 4 details our framework. Finally section 5 concludes the paper.

2. BASICS OF INTUITIONISTIC FUZZY SETS

Here we give some basic definitions (Atanossov 1999), which are used in the next section.

Definiton 2.1: Consider a set E. An intuitionistic fuzzy set (IFS) A in E is defined as an object of the following form

$$A = \{(x, \mu_A(x), \nu_A(x)) | \, x \in E\}$$

where the functions

$$\mu_A : E \rightarrow [0, 1] \text{ and}$$

$$\nu_A : E \rightarrow [0, 1]$$

define the degree of membership and the degree of non membership of the element $x \in E$, respectively.

And for every $x \in E$,

$$0 <= \mu_A + \nu_A <= 1$$

Obviously, each ordinary fuzzy set may be written as

$$\{(x, \mu_A(x), 1 - \mu_A(x) | \, x \in E\}$$

Definition 2.2: The value of

$$\Pi_A(x) = 1 - \mu_A(x) - \nu_A(x) \quad (1)$$

is called the degree of non-determinacy (or uncertainty) of the element $x \in E$ to the intuitionistic fuzzy set A. This may cater to either membership value or non-membership value or both.

Definition 2.3 : Distance between Intuitionistic Fuzzy Sets one of the most popular distances (Atanossov 1999) between the two Intuitionistic Fuzzy Sets A and B in $X = \{x_1, x_2, x_3, ..., x_n\}$ is the normalized hamming distance which is defined as follows:

$$I_{IFS}(A, B) = 1/2n\sum_{i=1}^{n}(|\mu_A(x_i) - \mu_B(x_i)| + |\nu_A(x_i) - \nu_B(x_i)| + |\pi_A(x_i) - \pi_B(x_i)|) \quad (2)$$

3. RELATED WORK

E-commerce is being extended from selling and buying of products to selling and buying of services. The researchers are trying to solve the problem of service provider selection, for delegation of task using various ways. One of them is from variants of reputation system in which consumers rate the service providers and ratings are kept on central server which help others to decide upon that service provider (Sabater and Sierra 2002) but this is only applicable for closed systems where central registry is easy to maintain. Some of the authors (Lixin *et al*. 2008, Lihong *et al*. 2008) have considered the problem of selection of service providers as MCDM (multi-criteria decision-making) problem and made algorithm for them but for distributed systems, such as e-commerce, these traditional approaches may fail to work in complex scenarios. Most schemes in open systems work with trust among entities (Huynh 2004, Yolum and Singh 2005, Yu and Singh 2002). Sreenath *et al*. (Sreenath and Singh 2003) also used trust factor to select the service provider among several present in open system but using central registry for service providers. In our system, we have also used the trust factor among entities but no central registry is there for service providers. Instead agents in the network maintain information about the service providers known to them either directly or indirectly, rather than a single entity maintaining information about all the service providers.

4. A PERSONALIZED AGENT BASED REPUTATION SYSTEM FOR FARMERS

To save time and to get quality results, farmers delegate some of their tasks to service providers available in electronic communities but selection of a service providers in these electronic communities is far difficult than selection of a product because products have many tangible characteristics attached with them like hardness, color whereas services can not be counted and measured due to their intangible nature (Parasuraman 1985). Understanding this, a personalized agent based reputation system framework is proposed for job delegation in e-market by the farmers. Fig. 1 illustrates working of the proposed system for farmers.

Basic methodology consists of the following steps:

1. Query generation from the requirements of the farmer.
2. Recommendations generation for a farmer.
3. Trust level updation of trustworthy acquaintances.

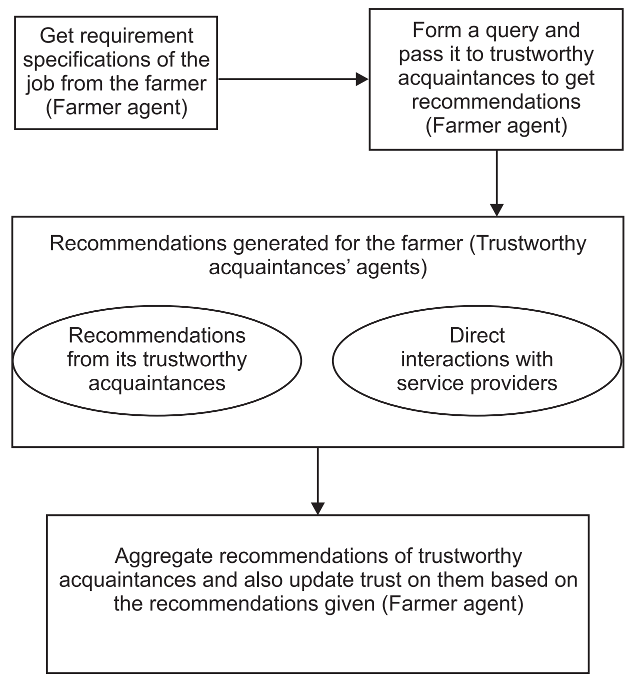

Fig. 1. Framework used to delegate the job by the Farmer

A. Query Generation from the Requirements of the Farmer

As in project development, requirement specification phase is very crucial so is this step. It is the farmer who being user of the system gives his requirement to his agent known as farmer agent. A service demand is expressed as well defined constraints on attributes of the service such as cost of service and completion time of service etc. farmer has to specify these attributes clearly to the agent so that job can be delegated to appropriate service provider. Year of establishment, location of the office of the service provider etc. many other factors are there which a farmer can specify, if required, in his service demand.

After getting the specification of the requirements from the farmer, farmer agent prepares the query in the form of request vector and sends this to its trustworthy agents to get the results. Later also this is the request vector that is passed whenever agent seeks recommendations from their trustworthy agents.

B. Recommendation Generation for Farmer

As in our society people generally use their own experience and take recommendations from their trustworthy acquaintances which act as referral recommendations. Similarly in our framework, agents exists in a network that acts as a society and follows social recommendation process (Gupta *et al*. 2011) For example if *A* trusts *B* and *B* trusts *C* and if *A* requests something from his trustworthy acquaintance *B*, *B* can also take help from his trustworthy acquaintance *C* to give recommendations. So, *A* has referral recommendations from those persons who are unknown to him i.e. *C*. In the presented work, at the time of request the farmer agent takes recommendations from those known to it, who may further take recommendations from those known to them and so on. Let x_{ij} $(i \neq j)$ represents the trust of agent *i* on agent *j* and x_{ji} represents trust level of agent *j* on agent *i.* Also, it is not necessary that $x_{ij} = x_{ji}$. Fig. 2 depicts this terminology i.e. if x_{12} represents trust level of agent 1 on agent 2 and x_{21} represents trust level of agent 2 on agent 1, then it is not necessary that $x_{12} = x_{21}$.

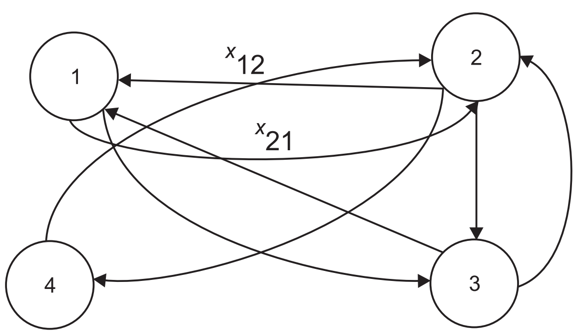

Fig. 2. Web of trust among agents

These trustworthy relationships form a web of trust (Bedi and Kaur 2005) which gives an advantage that agents are able to get the recommendations from even those agents that are unknown to it i.e. agent 1 can get recommendation from agent 4 also.

After getting the request from the farmer agent there are two ways that a trustworthy agent can generate the recommendations depending upon their relationship:

1. The agent has direct interaction with that service provider.
2. The agent comes to know about service provider through his trustworthy acquaintances.

The agents present in the system can act as service provider agents or as farmer agents depending on the requirements i.e. if they have services to supply then they will act as service provider agents and if delegation of the job is to be done then they will act as farmer agents. If the agent has already used the services of a service provider then that service provider will be a part of recommendation list only after comparing expertise vector of the service provider and the request vector (Yolum and Singh 2005). The preparation of the expertise vector of service provider is done on the basis of the attributes of the services he is providing. The farmer agent query can also be translated into request vector with the same dimensions of expertise vector (Gupta *et al*. 2010) After finding the cosine similarity between these two vectors, agents can conclude whether a service provider is capable to provide services to requesting farmer or not. Capability of the service provider can be calculated as in "(3)"

$$C = R \otimes E = \frac{\sum_{i=1}^{n} (r_i e_i)}{\sqrt{n \sum_{i=1}^{n} r_i^2}} \tag{3}$$

where $R(<r_1 \ldots r_n>)$ is a request vector and $E(<e_1 \ldots e_n>)$ is an expertise vector and *n* is the number of dimensions these vector have. To calculate $r_i e_i$ for parameters which are not binary, significance of the parameters is to be considered. For example, if requesting farmer can spend maximum (r_i) of Rs. 3500/- to get a service and service provider is providing the service @ (e_i) Rs. 2500/- then $r_i e_i = 1$ otherwise 0 i.e. if $r_i >= e_i$, then $r_i e_i = 1$ otherwise 0 whereas in case of experience of the service provider parameter, if requesting farmer wants that the service provider must have at least 5 years (r_i) of experience and the service provider has an experience of 10 years (e_i) then $r_i e_i = 1$ otherwise 0 i.e if $r_i <= e_i$, then $r_i e_i = 1$ otherwise 0.

The computed value of the capability of any service provider will help in selection of the service provider because if the selection is made only on the basis of reputation then delay in job completion may be introduced because many jobs may be delegated to the same service provider due to its good reputation and this may lead to delay in completion of the job. As it happens in real life that reputed shopkeepers are busy and take time in completion of the delegated job. So, if before delegation of the job apart from, capability of the service provider is also taken into account then it leads to the balanced system. Therefore, if $C > C_{\min}$, where $C_{\min}$ is the minimum value of the capability of the service provider desired to perform the job, then along with set of reputation of that service provider, capability of service provider will be computed and added to the list prepared by the agent. The set of reputation of service provider based on IFS contains membership, non membership and hesitation part. The membership part represented by $\mu_{reputation}(A, SP_j)$ can be calculated by "(4)" based on direct interaction of the agent with service provider as

$$\mu_{reputation}(A, SP_j) = \frac{\text{Directsuccessful}\ (A, SP_j)}{\text{Total}\ (A, SP_j)} \quad (4)$$

where $\mu_{reputation}(A, SP_j)$ represents the degree of membership of the j^{th} service provider (SP_j) in the set of reputation as computed by the agent A. Directsuccessful (A, SP_j) is the number of successful (satisfactory) interactions of the agent A with j^{th} service provider and Total (A, SP_j) represents the total number of interactions between agent A and j^{th} service provider.

As it is true when we talk with others in our idle time we get opinion about even those persons whom we have never met i.e. gaining knowledge about someone doesn't always require personal interactions with that person. However, we may not have full faith on this information but we cannot totally ignore it also, what if the information is correct. In our system, agents also show this behaviour by exchanging information about service providers in their idle time. Because there is an uncertainty in the information, we can model this information through IFS using its uncertainty part i.e. if the agent comes to know about a service provider through the informal talks then that information is mapped to $\pi_{reputation}(A, SP_j)$ and is calculated by "(5)" as

$$\pi_{reputation}(A, SP_j) = \frac{\sum_{i=1}^{m} \text{Trustlevel}_i * \dfrac{\text{Directsuccessful}\ (A_i, SP_j)}{\text{Total}\ (A_i, SP_j)}}{\sum_{i=1}^{m} \text{Trustlevel}_i} \quad (5)$$

$\pi_{reputation}(A, SP_j)$ represents the degree of uncertainty of the service provider SP_j. Trustlevel$_i$ represents the degree of trust on the i^{th} trustworthy acquaintance and is computed as shown in update procedure. Directsuccessful (A_i, SP_j) is the number of successful (satisfactory) interactions of the agent i with j^{th} service provider. Total (A_i, SP_j) represents the total number of interactions between agent i^{th} and the j^{th} service provider. m is the total number of acquaintances of the agent A.

Degree of non membership of SP_j represented by $\nu_{reputation}(A, SP_j)$ can easily be calculated as

$$\nu_{reputation}(A, SP_j) = 1 - \mu_{reputation}(A, SP_j) - \pi_{reputation}(A, SP_j) \quad (6)$$

This list of IFS sets must be then combined with the lists provided by the trustworthy acquaintances of the agent. For combining the lists, it is necessary to find degree of agreement of various trustworthy acquaintances for service providers. So, the requesting agent computes the normalized hamming distance, L between these IFS sets in lists as prepared by trustworthy acquaintances by "(7)".

$$L = \frac{\sum_{i=1}^{m-1} \sum_{j=i+1}^{m} \left(\left|\mu_{reputation}(A_i, SP_j) - \mu_{reputation}(A_i, SP_j)\right| + \left|\nu_{reputation}(A_i, SP_j) - \nu_{reputation}(A_i, SP_j)\right| + \left|\pi_{reputation}(A_i, SP_j) - \pi_{reputation}(A_i, SP_j)\right| \right)}{m(m-1)} \quad (7)$$

where L represents normalized hamming distance and m represents the total number of trustworthy agents who have responded. $\mu_{reputation}(A_i, SP_j)$ is the degree of membership of SP_j to the set reputation according to the agent A_i. $\nu_{reputation}(A_i, SP_j)$ is the degree of non membership of SP_j to the set reputation according to the agent A_i. $\pi_{reputation}(A_i, SP_j)$ is the degree of uncertainty of SP_j to the set reputation according to the agent A_i.

If the distance L between the set of reputations of any service provider is greater than the threshold, d_c, then the requesting agent will not take that service provider in the list because it shows that many peers do not have the same observation for that service provider but if they agree upon the reputation of any service provider then requesting agent will combine their set of reputations by computing the weighted average of the corresponding values of the recommendations of trustworthy acquaintances as given by equation "(8)", "(9)".

$$\mu_{\text{reputation}}(A, SP_j) = \frac{\text{Directsuccessful}(A, SP_j)}{\text{Total}(A, SP_j)}$$

$$\bigcup \frac{\sum_{i=1}^{m} \text{Trustlevel}_i * \mu_{\text{reputation}}(A_i, SP_j)}{\sum_{i=1}^{m} \text{Trustlevel}_i} \quad (8)$$

$$p_{\text{reputation}}(A, SP_j) = UI \bigcup \frac{\sum_{i=1}^{m} \text{Trustlevel}_i * \pi_{\text{reputation}}(A_i, SP_j)}{\sum_{i=1}^{m} \text{Trustlevel}_i} \quad (9)$$

where $\bigcup$ is the Fuzzy Union Operator, and *UI* represents the uncertain information acquired by the agent *A* about the service provider SP_j and is obtained from "(5)". $\mu_{\text{reputation}}(A, SP_j)$, $\pi_{\text{reputation}}(A, SP_j)$, Trustlevel_i all are same as in "(4)", "(5)". This procedure of aggregation is used by all recommenders and also by farmer agent.

Finally when this aggregated list of set of reputation of various service providers along with their capability value reaches farmer agent then only farmer agent computes the degree of significance of each service provider present in the list by "(10)" as

$$\text{DOS}_{SPi}(FA) = \mu_{\text{reputation}}(FA, SP_i) - (\nu_{\text{reputation}}(FA, SP_i) * \pi_{\text{reputation}}(FA, SP_i)) \quad (10)$$

$\text{DOS}_{SPi}(FA)$ represents degree of significance of i^{th} service provider i.e. SP_i in the aggregated list of service providers as prepared by the farmer agent. $\mu_{\text{reputation}}(FA, SP_i)$, $\pi_{\text{reputation}}(FA, SP_i)$ are aggregated values as computed by "(8)" and "(9)". This is a list prepared according to the reputation value of service provider. Taking this list and the list prepared according to the capability value of each service provider a final list can be prepared by "(11)"

$$\text{Rank}(SP_i)_{\text{final}} = \frac{w_1 * \text{Rank}(SP_i)_{\text{reputation}} + w_2 * \text{Rank}(SP_i)_{\text{capability}}}{w_1 + w_2} \quad (11)$$

where $\text{Rank}(SP_i)_{\text{final}}$ represents the position of SP_i i.e. i^{th} service provider in the list and $\text{Rank}(SP_i)_{\text{reputation}}$ represents position of SP_i in reputation list and $\text{Rank}(SP_i)_{\text{reputation}}$ represents position of SP_i in capability list.

Depending on the value of the timeout variable, the process of taking recommendations from trustworthy agents and preparing aggregated list takes place. The value of the timeout variable is given by the farmer depending upon the time interval for which he can wait to get result. Before further sending the request vector to his trustworthy acquaintances, to get recommendation, every agent checks timeout variable value which tells in how much time he has to give recommendations to requesting agent. Further sending of the request vector takes place only if the time is there to get the recommendation from his trustworthy acquaintances otherwise agent respond back with a list of set of reputation of service providers based on his knowledge to requesting agent.

C. *Updating Trust Level on Trustworthy Acquaintances*

Because trust is not a static quantity, it is very necessary to update the trust level on every source of recommendation (Gupta *et al*. 2009) The trust update can be done by computing the hamming distance between the sets of reputations as given by the trustworthy acquaintances and the aggregated set as computed by the requesting agent. Depending upon whether the distance between aggregated set and the recommended set is below an acceptable threshold d_t or not, the agent can update the trust level on their trustworthy agents using "(12)" assuming that d is the normalized hamming distance computed between two IFS to compute the difference in opinions

$$\text{Trustlevel}_i = \text{Trustlevel}_i + (d_t - d) \quad (12)$$

where Trustlevel_i represents trust level on i^{th} trustworthy acquaintance.

Hence, if there is agreement, $(d_t - d)$ is positive then Trustlevel on trustworthy acquaintance will increase but if there is difference in opinions then trust will decrease. The update process of the trust level is a continuous process and is done at every agent as soon as the aggregation of the list is done by the agent (Bedi and Kaur 2004).

Initially when the agents interact with other agents for the first time, then there is no previous information that the agent can use to define trust level on other agents. For such cases, initially the value of $Trustlevel_i$ can be set to 0 or to constant $c \in [0, 1]$. For example, when a very reputed organization starts a new sister concern, we do not consider its reputation as nil; rather we assign some minimum value to it (Sreenath and Singh 2004).

The advantage of above proposed framework gets restricted with social problems of the farmers because very less number of farmers are using internet for business (Ahmed and Kabir 2010) due to lack of infrastructure in rural areas. Efforts must be done to popularize usage and power of ICT in social and financial areas of farmers. In order to generate good results from the proposed system the farmer's text literacy limitations and the limited capability of the devices they use must be solved amicably (Warren 2004).

5. CASE STUDY

To illustrate use of the framework, we have considered eight service providers in a system providing an agriculture service of pest/weed control. We have also assumed that service demands given by farmers are to be specified as following four attributes i.e. Budget, state verification certificate, preferred method for pest control (biological/ chemical) and Experience. Table 1 shows sample data of two such farmers.

In our framework as each farmer is assisted by a farmer agent so the request of the farmer will first get translated into a query which will be getting fired to get the list of appropriate service providers. As an example we have considered the social network of six agents which are arranged as shown in Fig. 3 i.e. agent 1 has its trustworthy acquaintances as 2 and 3 and so on. Table 2 elaborates the Fig. 3 showing agents with their trustworthy acquaintances along with trust levels.

Each of the agents also maintains its knowledge about service providers along with their trustworthy acquaintances. Table 3 shows a sample of that information.

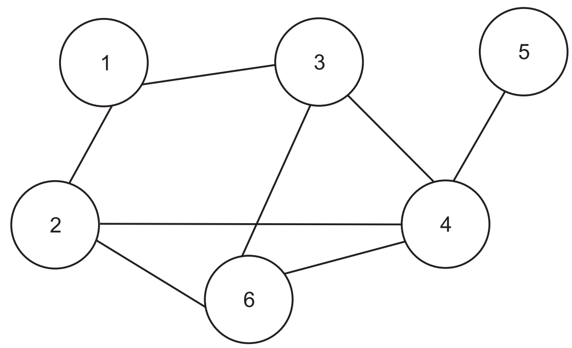

Fig. 3. Network of agents used for case study

Table 1. Ample of data as given by the farmers

Users	Budget	State verification certificate	Preferred method for pest control (biological/ chemical)	Experience
Farmer 1	max (Rs. 4000/-)	Should be there	Biological	min (5 yrs)
Farmer 2	max (Rs. 2000/-)	Not necessary	Chemical	min (2 yrs)

Table 2. Web of trust in tabular form along with trust levels that agents maintain on each other

Agents	Trustworthy acquaintances	Trust level
Agent 1	Agent 2	0.3
	Agent 3	0.9
Agent 2	Agent 1	0.5
	Agent 4	0.8
	Agent 6	0. 7
Agent 3	Agent 1	0.8
	Agent 4	0.4
	Agent 6	0.2
Agent 4	Agent 2	0.5
	Agent 3	0.4
	Agent 5	0.5
	Agent 6	0.8
Agent 5	Agent 4	0.5
Agent 6	Agent 2	0.6
	Agent 3	0.3

Table 3. Set of reputation of various service providers as computed by various agents

Agents → Service providers		Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Agent 6
X1	μ		0.5		0.3		
	ν		0.3		0.6		
X2	μ	0.2		0.3			0.7
	ν	0.3		0.5			0.1
X3	μ	0.4			0.6		
	ν	0.2			0.1		
X4	μ	0.6					
	ν	0.2					
X5	μ		0.2	0.5		0.5	
	ν		0.3	0.4		0.2	
X6	μ						0.2
	ν						0.1

Table 4. Capability of various service providers as computed by different agents.

Agents → Service providers	Agent 1	Agent 2	Agent 3	Agent 4	Agent 5	Agent 6
X1		0.3		0.4		
X2	0.5		0.1			0.3
X3	0.3			0.7		
X4	0.8					
X5		0.5	0.2		0.5	
X6						0.9

Empty cells in Table 4 represent that the agent mentioned in the column has no information about the service provider mentioned in the corresponding row.

As soon as the service demand reaches farmer agent 1, farmer agent passes this to agent 2 and agent 3, which are known to him, for their recommendations. Agent 2 and agent 3 compute the capability and reputation set of the service providers known to them taking into consideration service demand by "(3)" "(4)" and "(5)"and select some of the service providers. Then they pass the query to their trustworthy acquaintances to get recommendations. After getting the recommendations and aggregating those recommendations suppose agent 1 gets the aggregated recommendation list from agent 2 as {*X*1 (0.5, 0.3, 0.3), *X*5 (0.2, 0.3, 0.5)} and from agent 3 as {*X*2 (0.3, 0.5, 0.1), *X*5 (0.5, 0.4, 0.2)} where in *X*1(0.5, 0.3, 0.3), *Xi* represents i^{th} service provider. For example, 0.5 and 0.3 represent the aggregated reputation value of μ and ν, respectively; and 0.3 is the aggregated capability of *X*1 service provider given by agent 2. Trustworthy acquaintances of agent 2 and agent 3 can also get recommendations from agents known to them if timeout variable permits.

Now it is agent 1 who will aggregate the above recommendations received from agent 2 and agent 3 by "(8)" and "(9)" and are presented in Table 5.

Table 5. Aggregated list of service providers as perpared by agent 1

List of service providers	Aggregate set of reputation		Aggregate Capability
X1	μ	0.50	0.30
	ν	0.30	
X2	μ	0.24	0.59
	N	0.40	
X3	μ	0.40	0.30
	ν	0.20	
X4	μ	0.60	0.80
	N	0.20	
X5	μ	0.45	0.17
	ν	0.40	

Once the aggregated list of recommendations are there, the farmer agent (agent 1) then computes the degree of significance of each service provider present in the list by "(10)" as shown below in Table 6.

Farmer agent will now prepare a final list of service providers based on reputation and capability by "(11)" as shown in Table 7 which is supplied to the farmer for job delegation.

Table 6. List of service providers with their degree of significance as perpared by farmer agent

List of service providers	Degree of significance	Capability
X1	0.440	0.30
X2	0.004	0.59
X3	0.320	0.30
X4	0.560	0.80
X5	0.390	0.17

Table 7. List of service Providers supplied to farmer agent for job delegation

List of service providers	Rank
X1	0.8
X2	1.0
X3	1.0
X4	1.2
X5	2.0

6. CONCLUSIONS

A framework to solve the problem of selection of service provider to perform the job delegated by the farmer is proposed in this paper. In real life people use trust to safeguard themselves from false transactions. Similarly, if agents are replacing humans then they also have to learn how to employ trust to safeguard their interests. In our framework, agents form a social network and share experiences about service providers to help others. The decision of selection of service providers can easily be taken by evaluating these interactions. We have tried to do the same by preparing recommendations lists from trustworthy acquaintances based on capability and reputation of the service provider. The reason for taking capability attribute with reputation of any service provider is for reducing the time delay in accomplishment of the job because then the jobs may be distributed to maybe less reputable but capable service providers leading to more utilized and balanced system. Framework discussed in paper is applied to the agriculture scenario. It can help the big farmers/companies in selecting available service providers scientifically. This information can enhance their production and their income.

REFERENCES

Ahmed, A. and Kabir, L. (2010). An information plateform for low-literate villagers. *Proc. 24th Advanced Information Networking and Applications (AINA'10)*, 1271-1277.

Atanassov, K. (1999). Intuitionistic Fuzzy Sets: Theory and Applications. In: *Studies in Fuzziness and Soft Computing*, **35,** Physica-Verlag.

Bedi, P. and Kaur, H. (2004). Modeling agent reputation in uncertain environment. *Proc. International Conference on Cognitive Science*, Allahabad.

Bedi, P. and Kaur, H. (2005). Trust based recommender system. *Proc. Int'l Conference on AI*, 798-801, Las Vegas, USA.

Gupta, B., Kaur, H. and Bedi, P. (2010). A reputation based service provider selection system for farmers. *J. Inf. Assur. Secur.*, **5,** 508-515.

Gupta, B., Kaur, H. and Bedi, P. (2011). An agent based reputation system for unreliable grid environment. To be published in *proceedings of 3rd Int'l Conf. on Computer Modeling and Simulation* (*ICCMS*) held in Mumbai.

Gupta, B., Kaur, H. and Bedi, P. (2009). Trust based personalized ecommerce system for farmers. *Proc. 4th Int'l Conf. on Artificial Intelligence*, (IICAI), (2).

Huynh, T.D., Jennings, N.R. and Shadbol, N. (2004). Fire: An integrated trust and reputation model for open multiagent systems. *Proc. 16th European Conf. on Artificial Intelligence (ECAI'04)*, 18-22.

Lihong, M., Yanping, Z. and Zhiwei, Z. (2008). Improved VIKOR algorithm based on AHP and shannon entropy in the selection of thermal power enterprise's coal suppliers. *Proc. International Conference on Information Management. Innovation Management and Industrial Engineering* ICIII, 129-133, doi: 10.1109/ICIII.2008.29.

Lixin, D., Ying, L. and Zhiguang, Z. (2008). Selection of logistics service provider based on analytic network process and VIKOR algorithm. *Proc. IEEE International Conference on Networking, Sensing and Control (ICNSC '08)*, 1207-1210.

Parasuraman, A., Zeithaml, V.A. and Berry, L.L. (1985). A conceptual model of service quality and its implications for future research. *Jour. Market.*, 41-50.

Sabater, J. and Sierra, C. (2002). Reputation and social network analysis in multi-agent systems. *Proc. first Int'l Joint Conf. on Autonomous Agents and Multi-Agent Systems* (*AAMS'02*), 475-482.

Sreenath, M. and Singh, M.P. (2004). Agent based service selection. Web Semantics: Science, Services and Agents on the World Wide Web, **1(3)**, 261-279, Elsevier, 2004, doi:10.1016/j.websem.2003.11.006.

Warren, M.F. (2004). Farmers online: Drivers and impediments in adoption of internet in UK agricultural businesses. *Journal of Small Business Enterprise Development*, **11(3)**, 371-381.

Yolum, P. and Singh, M.P. (2005). Engineering self-organizing referral networks for trustworthy service selection. *IEEE Trans. Systems "Man and Cybernetics, part A*: *Systems and Humans*" **35(3)**, 396-407.

Yu, B. and Singh, M.P. (2002). Emergence of agent-based referral networks. *Proc. First Int'l Joint Conf. on Autonomous Agents and Multi-Agent Systems* (*AAMAS'02*), 1268-1269.