Using Hierarchical Bayesian Models to Learn About Reputation

Tracking Number 128

1. ABSTRACT

This paper addresses the problem of learning with whom to interact in situations in which obtaining information about others is associated with a cost, and this information is potentially unreliable. It considers a setting in which agents decide whether to enter a series of ventures with potential partners of unknown competencies. Agents can purchase reports about the competencies of partners from others, but the reliability of this information is unknown. A hierarchical Bayesian model was used to represent probabilistic information about partner agents and to infer the reliability of the purchased reports. Use of this model allows for simultaneous learning about the characteristics of individual partners as well as about the general pool of potential partners. The performance of this model was tested in experiments of varying complexity, and measured the benefits of using the model to make decisions as well as the effects of the cost of reputation information on agent performance. Agents using the model were able to learn to identify those information providers that were reliable, even when there was a high ratio of unreliable information providers.

2. INTRODUCTION

Trust and reputation are among the key driving forces of many economic systems, and have been studied in a variety of disciplines [8, 6, 11, 12]. This paper focuses specifically on the use of reputation information for learning whether to interact, and with whom, in an uncertain environment. By "reputation information", we mean opinions and beliefs about agents' capabilities as they are reported by other agents. Agents do not know the capabilities of other agents, but every agent is affected by the capability of the partner it chooses. An example of reputation information being used in the real world is that of an employer who seeks references of previous employment for job applicants to gain a better understanding of that applicant's capabilities.

We consider settings in which the acquisition of reputation information incurs a cost and the reliability of this information is uncertain. To make appropriate choices in such settings, agents must weigh the trade-offs between paying to obtain possibly unreliable reputation information and using their own experience to make a decision. For example, references may provide valuable information about the capabilities of job applicants, but to obtain this information employers will need to spend time and other resources. Some references may provide a biased description of the applicant, for example, to "boost" the applicants' prospects.

The setting we propose, called the venture domain, was inspired by examples from the world of financial markets, in which firms choose whether to invest in companies associated with a risk.[1] In the venture domain, a principal agent needs to decide whether to enter a series of ventures with different partner agents. Each interaction occurs over a sequence of finite, but unknown number of ventures with the same partner. The result of each of these ventures stochastically depends on the competence of the partner agent. Neither the competence of a particular partner nor the distribution over partner agents in general is known to the principal agent. Information about partners is accumulated by the principal agent through its direct interaction with them as well as by purchasing reports about the competency of partners from other agents. This information has the potential to aid the decision-making of the principal agent. For example, a principal agent that is informed that a potential partner is incompetent may disregard its own successful interaction with that partner as a "fluke".

The challenges of this domain arise from the fact that information about the true competencies of partners and the way in which they are assigned to the principal agent are never revealed. To make good decisions, agents can rely only on their own experience or the reports provided by others. To meet these challenges we make use of a hierarchical Bayesian model [4, 3]. These models have recently been used in several applications of AI and cognitive science for representing information at several levels of abstraction. In the venture domain this model explicitly represents and learns about the competency of partners in the world as well as the reliability of reputation providers. Using a hierarchical Bayesian model provides the following benefits: First, it allows principal agents to simultaneously learn about the competencies of individual partners as well as the distribution over the general pool of partners. Second, it combines information obtained from interacting in the world with background knowledge of principal agents which facilitates their learning about partner agents and the reliability of reputation providers. Lastly, it can naturally be coupled with a decision-making paradigm in which the policies of principal agents are based on their beliefs as inferred by the Bayesian model.

This model was evaluated in a variety of simulations that varied (1) the reliability of reputation providers' reports; (2) the number of possible ventures for a principal and partner agent pair; (3) the cost of reputation information; and (4) the complexity of the hierarchical model. We measured performance in two ways: the error in learning about agents' competencies, and the total reward obtained

Cite as: Title, Author(s), *Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, Decker, Sichman, Sierra and Castelfranchi (eds.), May, 10–15, 2009, Budapest, Hungary, pp. XXX-XXX.

[1]This example is somewhat disheartening given the developing international crisis in financial markets. However our model is not specific to finance and can be used in other areas in which there is inherent uncertainty about information.

over time for a principal agent using the model to act in the world. Agents using the hierarchical model were able to learn to identify those reputation providers that were more reliable, and to mesh the reputation information with their own experience in order to correctly predict the competencies of partner agents. For the case in which reputation information was very costly, the principal learned to purchase the minimal amount of information necessary to allow it to do better than without reputation information.

2.1 Related Approaches

Our work is distinguished from past approaches for learning about reputation in that we consider settings in which no information about the world is revealed to agents, and the process of obtaining reputation information is associated with a cost. Prior work has only investigated these issues in isolation.

Settings in which direct information about the world is available to agents includes the Agent Reputation and Trust test bed (ART) [2]. This test bed has spawned numerous models that have focused on learning from different sources of information providers in addition to personal experience [13]. In the competition, appraiser agents must estimate the values of different paintings, and may purchase information about these paintings as well as information about other appraisers. The distribution over paintings for appraisals is known to all agents, and the value of the painting is revealed after each interaction.

Several works consider settings in which the acquisition of reputation information is not associated with a cost. Fullam and Barber [1] proposed a model free approach for learning to weigh between reputation sources and personal information for the case in which there is a single source of reputation information. Their model is able to learn about particular agents, but not about agents in general. In the results section, we show how using this model to make decisions in our domain cannot capture a diverse population of partner agents.

Teacy *et al.* [14] consider a setting in which there are reputation providers of differing levels of reliability. Reece *et al.* [10, 9] consider an extended scenario where reputation information consists of multiple, possibly correlated attributes. In these works agents are allowed to learn about reputation from all possible information sources simultaneously, and do not consider the cost-benefit analysis associated with the decision to obtain reputation information. In contrast, in our setting acquiring reputation information is associated with a cost, and agents need to decide not only from whom to purchase information, but also whether it is worthwhile to purchase. We consider the interaction between cost and uncertainty, and their affects on performance.

3. THE SURROGATE VENTURE DOMAIN

The surrogate venture domain consists of interactions between a principal agent and a set of partner agents with unknown competencies. Each interaction N between a principal agent p and a partner agent w_θ encompasses a series of finite, unknown number of rounds. At each round, p can choose to enter a venture with w_θ for a cost c; this venture has an associated risk, for which it may receive a reward r_s, that depends on the agent's competency θ. The principal has the option to opt out at any round and be assigned to another partner agent for that round.

At the onset of each interaction, p may choose to buy reputation information about w_θ from one of M information providers for a query cost q. The reputation information from a chosen provider $m \in M$ consists of a pair (n_s, n_f) of successes and failures representing m's opinion of its interaction with w_{θ_t}. The reliability of this report is not known to p. The specifics of the set-up are as follows.

At each interaction N:
(1) A principal agent p is paired with partner agent w_θ.
w_θ has unknown competence θ
(2) Principal p can choose to purchase reputation information from provider m at a cost of q
(3) At each round in N:
(3.1) p can choose to opt out and go to step (1).
(3.2) otherwise, p enters a venture with w_θ at a cost of c.
(3.2.1) If the venture succeeds (with probability θ) then p incurs reward r_s.
(3.2.2) Otherwise, p obtains no reward.

4. A HIERARCHICAL MODEL OF INTERACTION

A hierarchical model for the venture domain is shown in Figure 1. The model describes three levels of abstraction. The bottom level contains the observations X obtained by the principal agent when interacting with a given partner agent over time. The competency of the partner agent is sampled from a Beta distribution with parameters (α, β) and makes up the mid-level of abstraction in the model. This level describes the principal's prior knowledge about the expected number of successful and failed ventures when interacting with the partner agent. The top level of the model describes the principal's prior knowledge about the way partner agents are distributed in the world. This is represented by a γ hyper-parameter that defines a Dirichlet distribution over a set of possible partner agent types. Each type induces a distribution over the competency of the possible partner agents, represented at the middle level of the model. For example, a type specifying a highly competent agent would have a high α count and a low β count.

Figure 1: A Hierarchical Bayesian Model of the Venture Domain

Given a value for the parameters of the distributions in the hierarchical model, we can construct a Bayesian network [7] to "zoom in" and describe a single interaction from the point of view of a principal agent p. This network is shown in Figure 2. Each node in the network represents a variable in the world and contains a local probability distribution over its domain given its parents in the network. The benefit to using a Bayesian network for this task is that its structure captures the conditional independence relationships that hold in the venture domain. We shall soon see how this facilitates inference with the model.

The node T in the network represents a possible type of partner agent. Each partner type $t \in T$ induces a partner agent w_{θ_t}. In our model, this means that the probability of a successful venture between p and w_{θ_t} is θ_t. The distribution over T represents the principal's beliefs about the competencies of the general population of partner agents; the distribution over θ_t represents the principal's beliefs about the competencies of an individual partner agent w_{θ_t}.

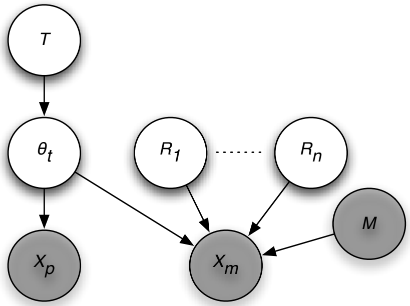

Figure 2: A Bayesian Network Modeling one Interaction in the Venture Domain. T is a discrete set of partner agent types; θ_t denotes the competence of agent $t \in \mathbf{T}$; X_p denotes the personal experience of the principal p; X_m denotes reputation information from provider m. Observations are represented as shaded nodes.

The node $X_p = (n_s, n_f)$ is a pair representing the number of successful and failed ventures in N observed by the principal agent for a single interaction with w_{θ_t}. The value of θ_t is determined by t but neither is observed by p. This is because agents' true competencies are never revealed to the principal agent.

At the beginning of each interaction, p may obtain reputation information about w_{θ_t} from one of K reputation providers. The Bayesian network is not directly used to choose between providers, but it represents the information that is obtained form each provider that is chosen. Each reputation provider agent j has a node R_j that represents the reliability of the information it provides. The node M denotes the identity of the reputation provider agent chosen by p. Its domain ranges over the K possible reputation providers. When $M = j$, this means that p has purchased a report from provider j, specified by the node $X_m = (n_s, n_f)$. This node contains an interaction history with w_{θ_t} according to the opinion of provider j. In the case that p has chosen not to request reputation information, both nodes M and x_m will equal null.

In theory, any model of the behavior of a reputation agent can be used in the model. For simplicity, we use two kinds of reputation agents: a reliable provider reports an interaction with the true partner w_{θ_t} and an unreliable provider reports a noisy interaction. We now show how to represent this in the Bayes net of Figure 2. We define each node R_j to be a Boolean variable. The parents of X_M are the nodes M and $R_1, \ldots, R_k$. The local probability model of X_M depends on these nodes as follows:[2] Let $M = j$ denote the identity of the provider chosen by p. If R_j is true, then X_M will be sampled from a series of ventures with w_{θ_t}. This represents a report from a truthful information provider. If R_j is false, then X_M will be sampled by a series of random coin tosses. This represents a report from a non-truthful provider.

Lastly, the Bayesian network is informed by the parameters of the Hierarchical network of Figure 1. The local probability model for T is a discrete distribution. The prior probability over T is a Dirichlet whose parameters are specified by γ_t in the hierarchical model. For each $t \in T$, θ_t is drawn from a Beta distribution with parameters α_t and β_t. The shaded nodes in Figure 2 represent those nodes that are observed by p in the interaction. These include the venture outcomes of the principal and the partner agent, the reputation information about the partner, and the identity of the reputation provider. The principle agent will need to use the network in order to learn the local probability distribution over the unobserved node θ_t.

4.1 Inference in the Model

In this section we show how a principle agent can use the network of Figure 1 to reason about particular partner agents as well as the way in which partner types are distributed in general. For this section we will assume that parameter values exists for both the Bayesian network of Figure 2 and the hierarchical model of Figure 1. Given a type t of a partner agent and a reputation provider m, the likelihood of the observation (x_p, x_m) depends on R_m and the reliability of m, which is unobserved. Because R_m is not observed, we need to sum over each of its possible values.

$$P^N(x_p, x_m \mid t, m) = \sum_{r_m} P^N(x_p, x_m, r_m \mid t, m) \quad (1)$$

By definition, each node in the Bayesian network is conditionally independent of its non-descendants given its parent nodes in the network. So, we can rewrite Equation 1 as the product of the venture outcomes of the principal X_p with the reputation information X_m provided by m, given that the reliability of the information is r_m, and the true partner type t.

$$P^N(x_p, x_m, r_m \mid t, m) = P^N(x_p \mid t) \cdot P^N(x_m \mid m, r_m, t) \cdot P^N(r_m \mid m, t) \quad (2)$$

The likelihood $P^N(x_p \mid t, m)$ depends on the unobserved partner competence θ_t. We integrate over this variable to get that

$$P^N(x_p \mid t) = \int_{\theta_t} P^N(x_p \mid t, \theta_t) \cdot P^N(\theta_t \mid t) d\theta_t \quad (3)$$

Using the network again, we can see that the node X_p is conditionally independent of T given θ_t, so we can replace $P^N(x_p \mid t, \theta_t)$ with $P^N(x_p \mid \theta_t)$.

Now, according to our model the type t deterministically determines θ_t, and θ_t follows a Beta (α_t, β_t) distribution. The distribution $P^N(x_p, t)$ is therefore a beta binomial with parameter θ_t:

$$P^N(x_p \mid t) = \binom{n_s + n_f}{n_s} \left(\frac{\alpha_t}{\alpha_t + \beta_t}\right)^{n_s} \left(\frac{\beta_t}{\alpha_t + \beta_t}\right)^{n_f} \quad (4)$$

The above equation allows us to define a local probability model for the node X_p representing the direct experience of p with partner w_{θ_t}. We can use the same distribution to model node X_m, representing the reported opinion of reputation provider m, for the case in which $R_m = true$ and $M = m$.

5. LEARNING

In this section we show how the principal agent can use both reputation information as well as its own experience to learn about the world. Because hierarchical Bayesian models allow information to be represented at several level of abstractions, we can use the past observations of the principal agent to update the model parameters at each level of abstraction.

5.1 Learning About Particular Partner Agents

We first show how a principal agent can adapt its beliefs to a particular partner agent after each venture in an interaction. This is done by using the past observations of the principal agent to update the local probability distributions in the Bayesian network of Figure 2. Let P^{N_i} denote the distribution over the network at the ith

[2]In the probabilistic reasoning literature, X_M is referred to as a multiplexer node.

venture of interaction N. The observations of the principal agent at the ith venture includes the outcome of venture, denoted o_i and the reputation information X_m obtained from provider m about the partner w_{θ_t}.[3]

We need to compute $P^{N_{i+1}}(t \mid o_i, x_m, m)$, the posterior probability of partner t given the observations incurred at the ith venture of interaction N. Using Bayes rule, we expand this probability to get that

$$P^{N_{i+1}}(t \mid o_i, x_m, m) \propto P^{N_i}(t \mid m) \cdot P^{N_i}(o_i, x_m \mid t, m) \quad (5)$$

The factor $P^{N_i}(t \mid m)$ is the prior distribution over the type space of partner agents at the ith venture. As can be seen in the network, this distribution does not depend on the node M. Therefore, we can replace the term $P^{N_i}(t \mid m)$ with $P^N(t)$, which is the local probability distribution of this node in the network.

Because the type t induces a partner of competency w_{θ_t}, we can infer the value of the node θ_t. This allows us to deduce from the network that the venture result o_i is conditionally independent of the reputation information x_m given t. So we can write

$$P^{N_i}(o_i, x_m \mid t, m) = P^{N_i}(o_i \mid t, m) \cdot P^{N_i}(x_m \mid m, t) \quad (6)$$

The factor $P^{N_i}(o_i \mid t, m)$ is the result of the ith venture, with partner w_{θ_t}. This probability does not depend on m and is drawn from a Bernoulli distribution with parameter θ_t. The probability $P^{N_i}(x_m \mid m, t)$ represents the likelihood of receiving report X_m from provider m given a partner θ_t. We compute this probability by summing over R_m, the reliability of provider m, in a way that is similar to that described in Equation 1.

5.2 Learning the General Population of Partner Agents

We now show how to use the observations obtained by the principal agent at each interaction to maintain a belief distribution over the general pool of partners. This is done by updating the parameters of the abstract layer of the model shown in Figure 1. These updates are performed once for every interaction. We first show how to compute the parameter values for the posterior distribution over θ_t by using the fact that the beta distribution of α_t^{N+1} is a conjugate prior to the binomial distribution over X_p. The updated value for α_t^{N+1} depends on the number of successful ventures incurred with partner of type t at interaction N, weighted by the likelihood that the principal agent was interacting with type t. The updated value for β^{N+1} is similar, but depends on the number of failed ventures.

$$\begin{aligned} \alpha_t^{N+1} &= \alpha_t^N + P^N(t \mid x_p, x_m, m) \cdot n_s \\ \beta_t^{N+1} &= \beta_t^N + P^N(t \mid x_p, x_m, m) \cdot n_f \end{aligned} \quad (7)$$

To compute $P^N(t \mid x_p, x_m, m)$ we expand using Bayes rule and use Equation 1 to compute the likelihood $P^N(x_p, x_m \mid t, m)$ in the Bayesian network of Figure 1.

In a similar fashion, we show how to update the prior distribution for T after each interaction N. Here we use the fact that the Dirichlet distribution of γ_t^{N+1} is a conjugate prior to the multinomial distribution of $P(T)$. The updated value for γ_t^{N+1} is

$$\gamma_t^{N+1} = \gamma_t^N + P^N(t \mid x_p, x_m, m) \quad (8)$$

5.3 Learning the Reliability of Providers

In general, the information reported by reputation providers may be correlated, and should be represented as a joint distribution. For example, if reputation agents are completely truthful, obtaining a report X_m about a partner agent w_θ increases the likelihood that agent m' will also report X_m when queried about w_θ. Modeling the correlation explicitly between any two reputation agent is difficult. However, we observe that at any given interaction N, only the chosen reputation provider m is "active", the identity of m is known to principal p and m is providing information generated from a single partner agent. Using the conditional independencies encoded in the Bayesian network, we can see that any two nodes r_m and R_j are conditionally independent on each other given T. Intuitively, this means that if we know the partner agent type and the identity of the reputation agent, its truthfulness does not depend on any other reputation agent. Therefore, we need only maintain $P(r_m \mid T)$ for each reputation agent and type.

To update the distribution over the truthfulness of each provider r_m, we use Bayes rule. This integrates the belief over r_m with the likelihood of the observations provided by m at round N.

$$\begin{aligned} P^{N+1}(r_m \mid x_p, x_m, m) = \sum_t & P^N(t) \\ & \cdot P^N(r_m \mid t, x_p, x_m, m) \end{aligned} \quad (9)$$

It is important to note that the nodes r_m and t are *not* independent of each other given X_m. Intuitively, we can gain knowledge about a partner's type knowing the reputation information provided by a reputation agent and whether or not it is providing reliable information. Therefore, we decompose $P^N(r_m \mid t, x_p, x_m, m)$ as follows

$$\begin{aligned} P^N(r_m \mid t, x_p, x_m, m) = & P^N(r_m \mid t) \\ & \cdot P^N(x_p, x_m \mid r_m, t, m) \cdot P^N(t) \end{aligned} \quad (10)$$

where the probability of the observations $P(x_p, x_m, m \mid r_m, t)$ is given in Equation 2.

6. DECISION-MAKING

To make good decisions with this model, we need to reason about the effects of present actions on the future. For example, it may be beneficial for the principal to reject an agent whose inferred competence is high, because agents with even higher competence are more likely in future. It is not possible to use backward induction type reasoning for this purpose, because the number of future rounds is not known in advance; also, updating and representing the belief state for every possible action by the principal is prohibitively difficult.

We therefore used a reinforcement learning paradigm in which the belief space is discretized to facilitate computation. The value of each state is determined by propagating the rewards and losses from its ventures during the course of its interaction with the partner. In this way, the benefit of interacting with partner agents in the past will inform the principal when it needs to make decisions in future rounds. The information the principal has gathered and analyzed in the past, influences its decision, because this information is embedded in the distribution over T, which is inferred by the hierarchical model. A similar approach is used by the principal agent when choosing to buy reputation information at the onset of a new interaction. There is a separate action for buying information from each reputation agent, including an action for choosing not to buy reputation information.

A crucial point is that the benefit of reputation information may only be appreciated after time. For example, when a truthful reputation agent reports many failed ventures for a particular partner, the principal agent may not interact with that partner, and its score for this interaction sequence will be zero. To be able to learn from

[3] o_i is essentially one element within the set X_p, and equals "success" or "failure".

such examples, we used a Boltzmann exploration policy, which encouraged the principal to explore the space of reputation agents in early rounds of interaction. This exploration rate decreases exponentially with respect to the number of times the principal agent has chosen to use a particular reputation agent.

7. EMPIRICAL EVALUATION

We evaluated the hierarchical model in a battery of simulations of the venture domain that varied in the cost of reputation, the extent of reliable reporting by other principals, the maximum number of rounds within each interaction between principal and partner agent, and the number of types used by the hierarchical model. Each interaction followed the description in Section 3.

In all simulations, the reward r for a successful venture was constant, and set to 12 units, while the investment cost c for entering a venture was set to 8 units, a value slightly below the expected reward of a venture given an average partner.[4] Each simulation was run for 40,000 cumulative rounds, composed of interactions of 1 to 16 rounds each. We varied the number of reputation agents from 1 to 64. We compared the performance of a principal using the hierarchical model (referred to as a "hierarchical agent") with the following decision-making models. The "History" agent used a standard reinforcement learning technique to model the competence of partner agents. It maintained a Markov Decision Process (MDP) whose observed state relied on its observations, whether obtained by the agent or provided by reputation agents. The MDP included a separate action for obtaining reputation information from each reputation agent (including a null action for not obtaining any information). The "Omniscient" agent was given the competence of the partner agent it was interacting with, the distribution over all types, and the number of rounds in each interaction. Its choice whether to reject or accept a partner agent depended solely on its expected future reward. This decision was made once at the onset of each interaction. In the setting we consider, the decision of whether to accept or reject a partner agent given its competence is clear cut, and depends on a threshold competence θ^*. When $P(\theta < \theta^*)$, the partner is rejected by the agent because its competence is below the threshold, and the omniscient agent incurs the relative utility for V_d computed in $d-1$ future rounds, which is

$$(d-1)V_d \cdot P(\theta < \theta^*) \tag{11}$$

When $P(\theta > \theta^*)$, the partner is accepted by the agent, and the omniscient agent receives the expected reward associated with that competence. In this case, the expected utility for future rounds is

$$\int_{\theta=\theta^*}^{1} P(\theta) \cdot (\theta r - c) d\theta \tag{12}$$

Thus, we can compute the expected utility V_d for this agent for d rounds as follows.

$$V_d = (d-1)V_d \cdot P(\theta < \theta^*) + d \cdot \int_{\theta=\theta^*}^{1} P(\theta) \cdot (\theta r - c) d\theta \tag{13}$$

The threshold value θ^* is that for which the agent is indifferent between rejecting and accepting the partner agent, so we add that

$$\theta^* \text{ satisfies } (d-1)V_d = d \cdot (\theta r - c)$$

Lastly, the "Myopic" agent chooses to accept a venture when it leads to a higher expected utility, given the past history of successful and failed ventures.

We measured performance for each round as the average score for an agent for all rounds using a given decision-making model,

[4]This number was set to encourage risk averse agents to invest.

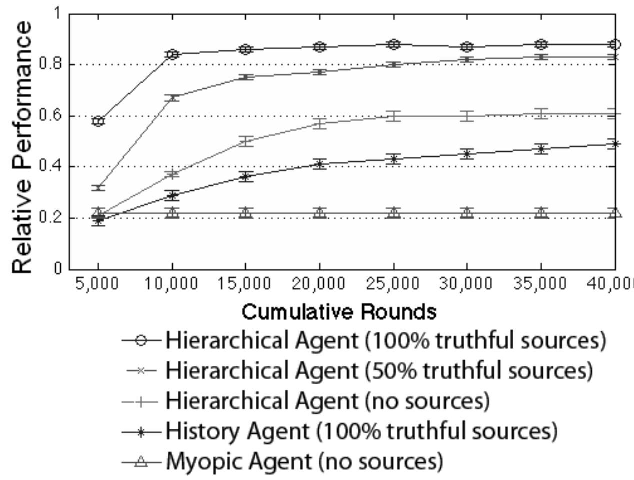

Figure 3: Performance of different agent models

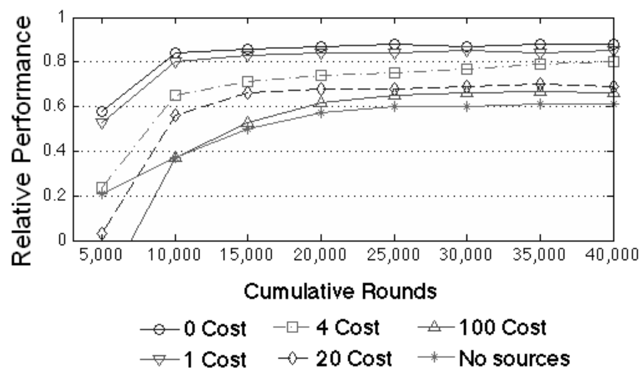

Figure 4: Performance of hierarchical agent model (varying costs of reputation sources)

normalized between an upper bound (the score obtained by the omniscient agent) and a lower bound (the score obtained by an agent that rejects all ventures). The following results relate to experiments where the maximum number of rounds per interaction was set at 16, and the number of types was set to two. An omniscient agent can compute Equation 13 in our domain simply by comparing the aggregate scores for d rounds of interaction with each of the two types.

Figure 3 illustrates the performance of the different agent models. When other reputation agents are reliable, the hierarchical agent achieved the highest performance. This performance decreased as non-reliable gossip agents were introduced, but still outperformed the other agent models. The Myopic agent, which has no look-ahead, achieved the lowest performance. We also compared performance to a recent model by Fullam and Barber [1], which use a separate reinforcement learning paradigm for each partner agent. This model achieves a constant performance of 0.63 in our setting (not shown in graph), and is outperformed by the hierarchical agent.

Figure 4 illustrates how the cost of obtaining reliable reputation influences the performance of the hierarchical agent. Even when information is expensive, only 10,000 rounds are needed before the performance exceeds the case where information is not available. This is because the hierarchical agent eventually learns not to buy costly information, while still exploiting the information it

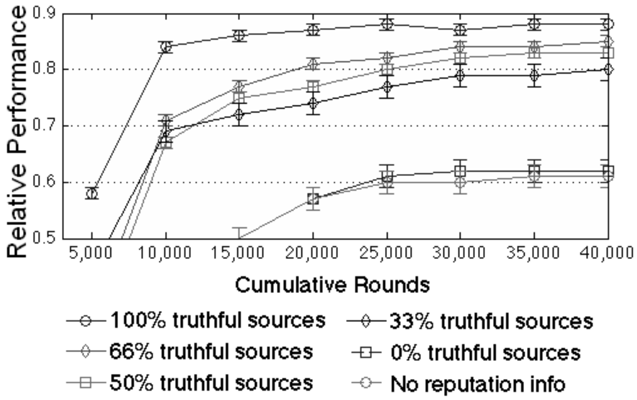

Figure 5: Performance of hierarchical agent (varying reliability of reputation sources)

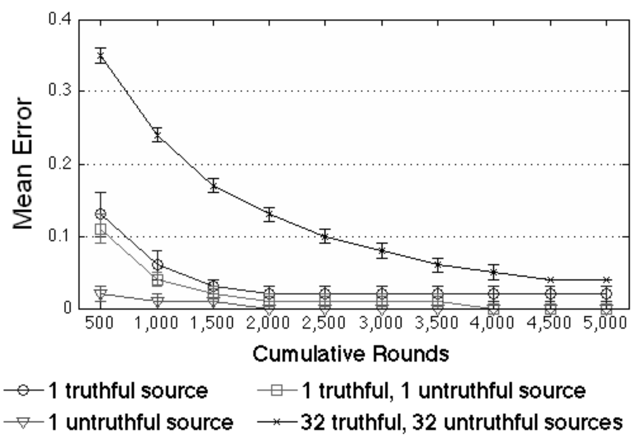

Figure 6: Model accuracy for reliability of reputation sources

purchased early on. Therefore, the initial gain of reliable reputation information can lead to a short term loss in performance, but will pay off in the long run. For the setting we used, the hierarchical agent stopped buying information when its cost was over 20 units.

Figure 5 shows the effects of varying measures of reliability among reputation agents on the performance of the hierarchical agent. As expected, when other reputation agents are completely reliable, the hierarchical model achieved the highest performance. As the ratio of reports from non-reliable agents begins to grow, the performance decreases, but the model still manages to maintain good performance. Even when reputation sources are only 33% reliable, the model is still performing within 80% of the optimal reward.

Figure 6-8 measure the accuracy of the hierarchical model. We list the average difference between the estimated and actual reliability of reputation agents in Figure 6. We list the average difference between the predicted partner competence and the actual partner competence in Figure 7. This evaluates the model's ability to learn about individual agents. We list the average difference between the estimated distribution over types representing partner's competence with the actual distribution in Figure 8. In all of these, the error was bounded by 0.05, even for the case in which reputation agents were only 50% reliable. The error is largest for the case in which there is no reputation information available. The speed in which the model learns is proportional to the ratio of reliable reputation agents. Figure 7 also includes a comparison with a model (denoted "Beta-reputation") that was used in prior work to learn about the competency of particular agents but not agents in general [5]. When reputation agents were always reliable, the average error of this model was 0.07, and was surpassed by the hierarchical model after 2,000 rounds. This work assumed reliability of reputation sources, and so we did not compare its performance in other conditions. The Hierarchical agents that do not have access to a reliable reputation agent estimate the types the most poorly. This happens for several reasons. First, they do not have any useful reputation information, and second, they are able to choose not to play with a partner agent after it has decided that the agent has a low competence. Because of this, the agent does not acquire an accurate estimation, it just has to estimate if the partner agent is a low performing partner and reject it.

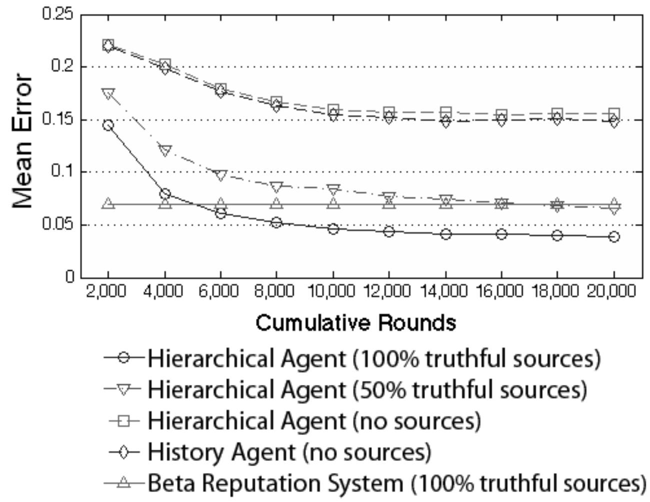

Figure 7: Model Accuracy for competence of partners

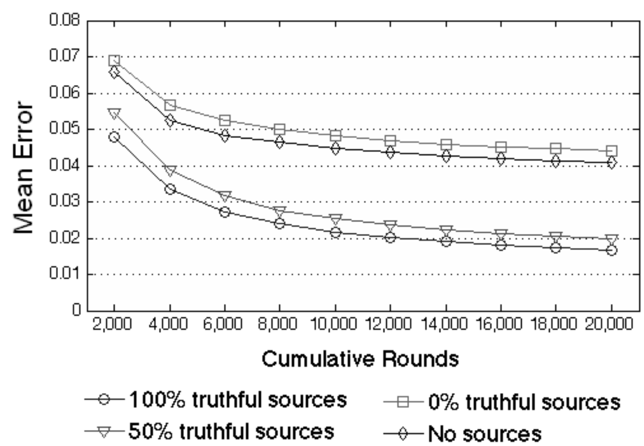

Figure 8: Model Accuracy for type estimation

Figure 7 plots the performance of the hierarchical agent in scenarios where there are up to 8 rounds per interaction. (This is in contrast to Figure 5 which plots the performance of the hierarchical agent for 16 rounds per interaction.) The average performance of the hierarchical agent decreases as the number of rounds per interaction decrease. This is due to the difficulty of determining the partner agent type in such a short period of time.

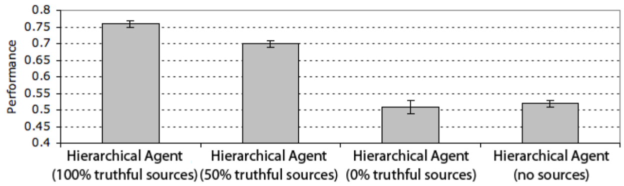

Figure 9: Performance for hierarchical agents using 8 rounds per interaction.

Lastly, we conducted experiments in which the number of real world types and the types used by the hierarchical agent to model those types were varied. When the number of modeled types were fixed to two, and the number of real world types varied, we found that the hierarchical agent received slightly higher performance as the number of real world types increased. The highest performance was achieved, as expected, when the number of types used in the model matched the number of types in the real world.

7.1 Discussion

There are several contributing factors to the success of the Bayesian hierarchical model. First, the model integrates learning about the competence of individual agents with that of agents in general. An agent using this model is thus able to make the right decision when paired with an individual partner for a short amount of time. To demonstrate, the number of rounds in each of our interactions were one or two orders of magnitudes less than those used by Fullam and Barber, yet our model was still able to learn better, or as well, as these. Second, the model separates the learning about the truthfulness of reputation agents from that of agents' competencies, which allows it to weigh the tradeoffs between using the existing model to make a decision and deciding to buy more information at a cost. Third, the model is able to learn the truthfulness of each information provider, allowing a principal to pick-and-choose the information that it deems most valuable.

The performance of all hierarchical models plateaued around 30,000 rounds. Interestingly, the performance of the History Agent monotonically increased. In a separate examination running 200,000 rounds, it was shown that when reputation agents were truthful, the performance of the History agent was similar to that of the hierarchical agent. However, as the ratio of non-truthful reporters increased, the performance of this agent dropped significantly below that of the hierarchical model for these settings.

8. CONCLUSION

This paper presented a Bayesian hierarchical model for modeling agent competence and the truthfulness of agents giving reputation information. Results show that the principal agent using the hierarchical model was able to accurately learn agents' capabilities and use this information to quickly estimate the competence of individual partner agents. The model was robust to inaccurate and potentially costly reputation information. In addition, we showed that short term loss in performance due to buying costly reputation information lead to long term increases in performance over using no reputation information at all. In future work, we will first wish to extend the complexity of the environment to the case in which non-truthful reputation agents are strategic, and may report malicious information. Second, we wish to explore the conditions in which multiple, possibly competing, principal agents share their learned information. Lastly, we intend to use this model to learn how humans both provide reputation information and learn about reputation.

9. REFERENCES

[1] K. Fullam and K. S. Barber. Dynamically learning sources of trust information: Experience vs. reputation. *Autonomous Agents and Multi-Agent Systems*, pages 1055–1062, 2007.

[2] K. K. Fullam, T. B. Klos, G. Muller, J. Sabater, A. Schlosser, Z. Topol, K. S. Barber, J. S. Rosenschein, L. Vercouter, and M. Voss. A specification of the agent reputation and trust (art) testbed: experimentation and competition for trust in agent societies. In *AAMAS '05*, pages 512–518, New York, NY, USA, 2005. ACM.

[3] A. Gelman, J. Carlin, H. Stern, and D. Rubin. *Bayesian data analysis*, pages 103–331. Oxford University Press, 2003.

[4] I. Good. *Bayesian Statistics: Some history of the hierarchical Bayesian methodology*, pages 489–519. AUAI Press, 2003.

[5] A. Josang and R. Ismail. The beta reputation system. In *Proceedings of the 15th Bled Conference on Electronic Commerce*, Bled, Slovenia, 2002.

[6] T. Khopkar, X. Li, and P. Resnick. Self-selection, slipping, salvaging, slacking, and stoning: the impacts of negative feedback at ebay. *Proceedings of the 6th ACM conference on Electronic commerce*, pages 223–231, 2005.

[7] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, 1988.

[8] S. Ramchurn, T. Huynh, and N. Jennings. Trust in multi-agent systems. *The Knowledge Engineering Review*, 19(1):1–25, 2004.

[9] S. Reece, A. Rogers, S. Roberts, and N. R. Jennings. A multi-dimensional trust model for heterogeneous contract observations. In *Twenty-Second AAAI Conference on Artificial Intelligence*, July 2007.

[10] S. Reece, A. Rogers, S. Roberts, and N. R. Jennings. Rumours and reputation: evaluating multi-dimensional trust within a decentralised reputation system. In *AAMAS '07*, pages 1–8, New York, NY, USA, 2007. ACM.

[11] J. Sabater and C. Sierra. Regret: reputation in gregarious societies. In *AGENTS '01: Proceedings of the fifth international conference on Autonomous agents*, pages 194–195, New York, NY, USA, 2001. ACM.

[12] S. Sen. Believing others: Pros and cons. *Artificial Intelligence*, 142(2):179–203, December 2002.

[13] W. T. Teacy, T. D. Huynh, R. K. Dash, N. R. Jennings, M. Luck, and J. Patel. The art of iam: The winning strategy for the 2006 competition. *The 10th International Workshop on Trust in Agent Societies*, 2007 2007.

[14] W. T. Teacy, J. Patel, N. R. Jennings, and M. Luck. Travos: Trust and reputation in the context of inaccurate information sources. *Autonomous Agents and Multi-Agent Systems*, 12(2):183–198, 2006.