Ladders of Success: An Empirical Approach to Trust

Pınar Yolum
pyolum@eos.ncsu.edu

Munindar P. Singh
singh@ncsu.edu

Department of Computer Science
North Carolina State University
Raleigh, NC 27695-7535, USA

Categories and Subject Descriptors
I.2.11 [Artificial Intelligence]: Distributed Artificial Intelligence

General Terms
Experimentation, Performance

Keywords
Trust

1. INTRODUCTION

We consider the problem of trust in open environments—large-scale, decentralized systems consisting of autonomous agents. The key problem is how an agent should trust another agent. Trust is for a purpose. That is, a truster would (or would not) trust a trustee for a particular service. We consider a setting wherein different agents consume and provide services to one another. The agents offer varying levels of trustworthiness and are interested in finding trustworthy agents who provide the services that they need.

It is helpful to consider how trust is conceptualized. Institutional trust or trust in authoritative institutions or organizations is common in the off-line world. People trust these institutions to stabilize their interactions. Current distributed trust management can be thought of as formalizing institutional trust, because it assumes that digital certificates issued by various certificate authorities lead to trust. In open environments, however, there may not be any central trusted authorities. Even if there are such authorities, for trust to develop, the other participants must somehow recognize them as authoritative.

Multiagent approaches take an empirical stance on trust, attempting to create trust based on evidence of some sort. The evidence could be local or social. Social trust means trusting an agent based on information from individual witnesses or from a reputation agency. The credentials of the witnesses or reputation agencies are crucial for interpreting this second-hand information correctly. Hence, the trustworthiness of the information sources must be established as well [2, p. 74]. Referrals are a powerful way of ensuring that the sources themselves are trustworthy [1]. Local trust means considering previous direct interactions. These are valuable—since the truster itself evaluates the interactions, the results are more reliable.

Previous approaches for trust emphasize either its local or its social aspects. By contrast, our approach takes a strong stance for both aspects. Here, the agents track each other’s trustworthiness locally and can give and receive referrals. This enables us to address two properties of trust that are not adequately addressed by current approaches. One, trust often builds up over interactions. That is, you might trust a stranger for a low-value transaction, but would only trust a known party for a high-value transaction. Two, trust often flows across service types. That is, you might assume that a party who is trustworthy in one kind of dealing will also be trustworthy in related kinds of dealings.

2. GRAPH-BASED REPRESENTATION

We consider a setting with a fixed number of service types. Service providers offer one or more of these services. Some of these services may be related, i.e., being a good provider for one may imply being a good provider for another. Conversely, some services may be unrelated to each other.

We represent the services as a graph whose nodes map to service types. The graph representation is more expressive than a vector representation because it can capture relationships between service types that a vector representation cannot. For example, a service provider that has been found to be trustworthy for one type of service can be considered for another type of service based on how well performance of the services correlates.

When an agent needs a provider for a service for which it knows of no providers, it can potentially ask others or promote a provider that it has used for another service. Promotions provide a systematic way to reuse previous experiences with the service providers. Figure 1 illustrates a setting with partially ordered services. Any two services that are related are joined by an edge. Here an edge \((s_i, s_j)\) indicates that a provider who can perform \(s_i\) well may also be able to perform \(s_j\) well. A provider is tried for a new service if it has performed well for another service, and if performing well in the first service indicates that the provider may perform well for the second service. The likelihood of a service provider in a lower node to perform a service in the upper node is represented by weights on the edges. For example, the weight 0.5 from \(S_0\) to \(S_1\) means that a provider of \(S_0\) will likely be providing \(S_1\) half the time.

Each agent maintains its own service graph to autonomously capture its experiences. Thus agents may have differing edges and weights for the same pair of services. The weights are adjusted...
independently by each agent. After delivering a service, a service provider is rated by the consumer. The rating reflects the satisfaction of the consumer. These ratings are used by the consumer to decide if this service provider will be used again or referred to other consumers. Service providers with low ratings are replaced with service providers that can potentially get higher ratings.

When promoting a provider from $s_i$ to $s_j$, two factors are considered: how trustworthy the provider is for $s_i$ and how well related $s_i$ and $s_j$ are. We calculate the trustworthiness of the provider $p$ at $s_i$ ($t_{ps}$) through its ratings at $s_i$ and the number of interactions (for $s_i$). The strength of the relation between $s_i$ and $s_j$ is given by the edge weight, $w_{ij}$.

The product of the edge weight with the average ratings $((w_{ij} \times t_{ps}) > \theta)$ projects how much the provider $P$ can reproduce its ratings in $s_j$. If this projected value is greater than a promotion threshold $\theta$, then $P$ can be promoted to perform $s_j$. In the extreme case, if $w_{ij} = 0$ (the services are not correlated), then the service provider is not expected to perform well in $s_j$ even if it performs well in $s_i$. Conversely, if a provider is not trusted for $s_i$ ($t_{ps} = 0$), then the provider will never be promoted to $s_j$ irrespective of how correlated the two services are.

The weights that denote the relation between two services are estimated by each agent, which can update the weights in its graph based on its experiences. Hence, two agents can have different weights for the same edge. The graph weights are updated after promoting a provider and testing it for the higher service. The graph weights are updated after each interaction between two services.

4. EVALUATION

We study the factors that affect finding trustworthy providers.

- **Initial Setting:** Trust prejudice denotes whether an agent is willing to trust new service providers. We capture this intuition through the initial graph weights. For example, if initially all the weights are 1, then the agents are willing to try out all new service providers in all types of services.
- **Promotion Threshold:** The estimated weight between two services is adjusted based on previous promotions between the two services. The higher the promotion threshold the less risk an agent is willing to take in its promotions.
- **Number of Interactions:** It is widely accepted that the number of previous interactions increases the accuracy of the trust assessment. We study the number of interactions as a prerequisite for promotions.

Essentially, each agent optimizes its performance by minimizing its promotion errors and maximizing the number of service providers it finds when desired (i.e., effectiveness). Our key finding is that there is a positive relation between promotion error and the effectiveness, such that if the consumers are cautious and promote reluctantly up the graph, they make fewer mistakes but might also miss many useful promotions, leading to sub-optimal effectiveness.

On the other extreme, with low promotion thresholds, the agents always find service providers they need, but make many promotion errors. Hence, optimal performance lies in the middle values of the promotion threshold. Among these, the performance is always better when the number of interactions is either 1 or 2. This suggests that the third interaction does not add much value to the performance. Further, when agents generate enough queries, the initial setting does not affect these results.

5. ACKNOWLEDGMENTS

This research was supported by the National Science Foundation under grant ITR-0081742. We thank the anonymous reviewers for helpful comments.

6. REFERENCES