Accounting for Circumstances in Reputation Assessment

(Extended Abstract)

Simon Miles
Department of Informatics
King's College London
Strand, London, UK WC2R2LS
simon.miles@kcl.ac.uk

Nathan Griffiths
Department of Computer Science
University of Warwick
Coventry, UK CV4 7AL
nathan.griffiths@warwick.ac.uk

ABSTRACT

Reputation, influenced by ratings from past clients, is crucial for providers competing for custom. For new providers with less track record, a few negative ratings can harm their chances of growing. Aside from malicious or subjective ratings, addressed in existing work, an honest balanced review of a service provision may still be an unreliable predictor of future performance if the circumstances differ. For example, while a delivery service may be generally reliable, a particular delivery may be delayed by flooding. A common way to ameliorate the ratings that may not reflect future performance is by weighting by recency. We argue that better results are obtained by querying records of how services are provided for patterns indicating the circumstances of provision, to determine the significance of past interactions.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence — *Multiagent systems*

General Terms

Algorithms, Experimentation

Keywords

Reputation, Trust, Provenance, Circumstances

1. INTRODUCTION

In service-oriented systems, an accurate assessment of reputation is essential for selecting between providers. Existing reputation models, such as FIRE [1], and HABIT [2], typically use a combination of direct and indirect experience, with numerical representations for reputation. Where there is little data from which to assess reputation, individual pieces of evidence carry great weight and may cause a provider to be rarely selected, so never given the opportunity to build their reputation. While the honesty of a reviewer can be tested from past behaviour and dishonest negative reviews ignored, it is possible for a review to be accurately negative, because a service was provided poorly, but for this not to be an accurate predictor of future behaviour. These are *mitigating circumstances*, where the temporary context of a past provision rather than the ability of the agent is the reason it was poorly provided. Most approaches use *recency* to ameliorate these effects, but this is a blunt instrument, as recent provision may have been affected by mitigating circumstances, and older interactions may remain good predictors of current reliability, because the circumstances are comparable. We outline a reputation assessment method based on querying detailed records of provision, using patterns describing past interaction circumstances to determine their relevance. By employing a standard model for describing these circumstances, we give a practical means by which agents can model, record and query the past.

2. APPROACH

We argue for the circumstances of past interactions to be recorded and taken into account explicitly. This raises the question of what form these records should take, and who should record them. Also, in order to share interaction records between agents, as many reputation methods do, they need to be recorded in a commonly interpretable format. In 2013, the W3C standards body published the PROV standard for modelling, serialising and accessing *provenance information*, the history of processes [3]. A PROV document describes in a queryable form the causes and effects within a particular past process of a system (such as agents interacting, the execution of a program, or enactment of a physical world process), as a directed graph with annotations.

Mitigation can have many forms, such as a subsequently replaced sub-contractor failing to deliver on time, or a client failing to specify required conditions (e.g. expiration date of goods being shipped). A reputation assessor looks for patterns in the provenance that indicate situations relevant to the current client's needs and mitigating circumstances affecting the assessment of providers. Provenance data is suitable for this because it includes the causal connections between interactions, and so captures the dependencies between agents' actions. It can include multiple parties to an interaction and their organisational connections, The assessor filters the provenance for key subgraphs from which reputation can be assessed using existing approaches, by identifying successful and failed interactions, and adjusting these by mitigation and situation relevance.

As an example, previous poor service by a provider could have been due to their reliance on a poor sub-provider for some aspect of the service. If the provider has changed sub-provider, the past interaction should not be considered relevant to their current reputation. In other words, Provider

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May 4–8, 2015, Istanbul, Turkey.*

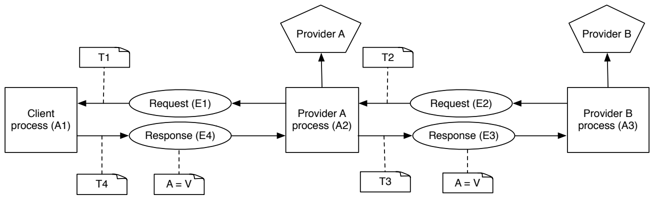

Figure 1: Provenance pattern showing sub-provider

A's reputation should account for the fact that previous poor service was due to Provider A relying on Provider B, who they no longer use. The provenance should show that Provider B was used where there was poor service provision, Provider B's activities were the likely cause of the poor provision, and Provider A no longer uses Provider B. A PROV pattern showing reliance on a sub-provider in a particular instance can be defined as shown in Figure 1. Activities are labelled with An (where n is a number) and entities are labelled with En. This encapsulates that *(i)* a client process, A1, sends a request, E1, for a service to a service process, A2, for which Provider A is responsible; *(ii)* A2 sends a request, E2, to a service process, A3, for which Provider B is responsible; *(iii)* A3 completes the action requested, and sends a result, E3, back to A2; *(iv)* A2 completes the service provision, sending the result, E4, back to A1, so that the client has received the service requested. Some interactions are labelled with timestamps (T1, T2...) and some entities with labels indicating a quality metric (A=V).

3. EVALUATION

We evaluated our approach through simulation, comparing it with FIRE. Reputation is assessed in FIRE from *rating* tuples of form (a, b, c, i, v), where a and b are agents that participated in interaction i such that a gave b a rating value of $v \in [-1, +1]$ for the term c (e.g. reliability, quality, timeliness). A rating of $+1$ is absolutely positive, -1 is absolutely negative, and 0 is neutral. FIRE gives more weight to recent interactions using a *rating weight function*, for each reputation type, where $K \in \{I, W\}$ representing the assessor's own past interaction ratings and those of witnesses respectively. (We do not include role-based and certified reputation as they are tangiential to the focus of this paper.) The trust value agent a has in b with respect to term c is calculated as the weighted mean of the available ratings, and the overall trust in an agent is calculated as a weighted mean of each of the component sources. Mitigating circumstances can be incorporated into existing reputation models by adjusting the weight given to each rating resulting from an interaction for which there are mitigating circumstances. In FIRE, this can be done through the rating weight function, for each type of reputation by a factor that accounts for mitigation. This factor should reflect how convincing an agent considers particular mitigating circumstances, defined per pattern. For the sub-provider pattern above this corresponds to the perceived contribution of the sub-provider to the provision.

We evaluated the strategies on a simulated network of 100 agents providing services to each other over 1000 rounds. Agents are positioned on, and explore, a spherical world which dictates their neighbours and acquaintances, with an average of 3 neighbours each. There were 5 primary capabilities (types of service which may require sub-capabilities), capabilities have two terms (quality and timeliness), and each agent has 3 capabilities. Each agent has a 50% chance to request a service each round and 20% chance not to pick the most trusted agent (so exploring the provider space). Agents switch sub-provider after a period of 1–15 rounds. Where recency scaling was applied, it was set such that after 5 rounds it is 50% weight. The utility gained in a round is the sum of utility gained per service provision, where the latter is the average of quality and timeliness of the provision (each in $[-1, 1]$). We compared five strategies: FIRE, our approach (Mitigating) with and without recency, FIRE without recency, and random selection. Each strategy was evaluated in 50 networks and the results averaged. Figure 2 shows the results where three circumstance patterns are present: poor sub-providers, freak events and poor organisational culture. Our approach has improved performance, both with and without recency, over FIRE, with an improvement of 10.1% without and 9.3% with recency scaling respectively. The recency scaling of FIRE is also shown to be beneficial where mitigating circumstances are not taken into account, i.e. FIRE is better than FIRE without recency. These results match the intuition that recency is valuable for taking account of changes in circumstances, but is crude compared to what is possible when past circumstances are visible. When recency is combined with mitigating circumstances there is negligible improvement.

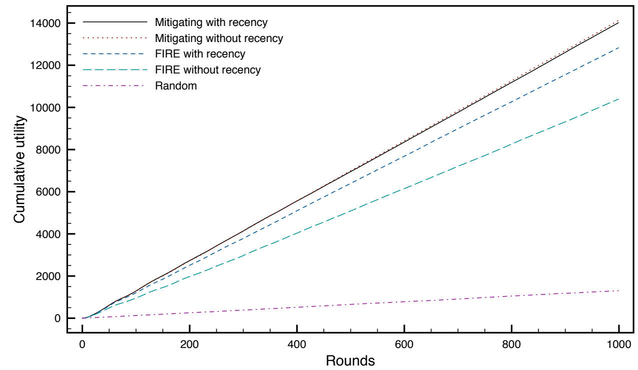

Figure 2: Comparison of various assessment models.

Acknowledgements

This work was part funded by the UK's EPSRC, under project JASPR, ref. EP/M012654/1 and EP/M012662/1.

REFERENCES

[1] T. D. Huynh, N. R. Jennings, and N. R. Shadbolt. An integrated trust and reputation model for open multi-agent systems. *J. of Autonomous Agents and Multi-Agent Systems*, 13(2):119–154, 2006.

[2] W. T. L. Teacy, M. Luck, A. Rogers, and N. R. Jennings. An efficient and versatile approach to trust and reputation using hierarchical bayesian modelling. *Artificial Intelligence*, 193:149–185, 2012.

[3] W3C. PROV model primer. http://www.w3.org/TR/prov-primer/, 2013.