Consiglio Nazionale delle Ricerche

A Client-Aware Reputation System for e-Services

G. Costantino, F. Martinelli, M. Petrocchi

IIT TR-11/2011

Technical report

luglio 2011

Istituto di Informatica e Telematica

A Client-Aware Reputation System for e-Services*

Gianpiero Costantino, Fabio Martinelli, and Marinella Petrocchi

IIT-CNR, Pisa, Italy
E-mail: `name.surname@iit.cnr.it`

Abstract. In the Internet age, people are becoming more an more familiar in experiencing online services. Given the intrinsic distributed nature of the electronic transactions involved, there is the need to prove somehow the trustworthiness of such services for supporting a user in her choice. To this aim, different techniques have been presented. For example, a simple but useful solution is to rely on feedback of past users testifying if they have been satisfied by a service.
In this paper, we consider a scenario for business transactions where a reputation management system helps clients in choosing services that best satisfy their attitudes and preferences. Specifically, a reputation value is associated to each service at stake. This value is updated according to past and new clients interactions. In fact, at the end of each interaction with a service, clients provide feedback regarding that service. The main feature of our proposal is the *client awareness*. This derives from designing and implementing a probabilistic client model based on real behaviours of users when they choose a service and give feedback. This client model has been obtained by collecting and processing real data from ones of the most popular websites for travel advice.
We present an evaluation aimed at validating our proposal. In the simulations, we also deal with the issue of *false feedback*, reported by clients that intentionally aim at subverting the reputation value of a service. The simulations results show that our system is robust up to a certain number of malicious feedback.

Keywords: Trust, client-model, feedback, broker, malicious-clients.

1 Introduction

The new concept of *aaS (Everything as a Service) [1,2,3] spans over specific domains for technicians, like the creation of software applications on the provider's remote platform (*e.g.*, GoogleApps and Force.com), or the provision of networking components and virtual servers to support storage operations, like, *e.g.*, Amazon Web Services. Also, software may be hosted and operated by enterprise vendors over the Internet. In such a way, customers do not need to buy software licenses or additional infrastructure equipment, since applications are not installed and do not run on their own computers.

However, with the support of new advanced technologies and standards, the design and deployment of e-services will become easier and no limited to specialised categories of users. This is the case, *e.g.*, of the upcoming infrastructure known as the Smart

* Work partially supported by the EU project FP7-257930 Aniketos: *Ensuring Trustworthiness and Security in Service Composition* and by the EU project FP7-256980 Nessos: *Network of Excellence on Engineering Secure Future Internet Software Services and Systems*.

Grid [4], that will enable real-time communication between a consumer and the utility providers (such as heat, light or electricity), allowing the consumer to optimize energy usage and help society getting the most out of the use of renewable energy sources.

The availability of a large pool of e-services, similar for functionalities but with different quality levels, may lead consumers to face the difficulty of choosing the one(s) that satisfy at best their needs and preferences. Moreover, the requested service could be, in its turn, composed of sub-services. As an example, an online travel agency could provide separately a set of accommodation services and a set of transportation services, and a consumer would like to book a packet including a hotel and a car rental that best meet her requirements. What generally helps in such situations is a service provider in charge of delivering a list of e-services required by the consumer, together with additional criteria supporting the consumer in her choice. A natural support is represented by the *reputation* of a service, typically a numerical value summing up the degree of satisfaction of past users towards that service. Nowadays, many websites, specialised in customer reviews, publish the reputation value of the advertised services, as the result of a set of user feedback. High reputation values will encourage the consumer in making her choice, even if the final selection will be influenced also by personal preferences (*e.g.*, users will not always choose the hotel with the highest reputation, as it is also probably one of the most expensive).

The contribution of this paper is to present a reputation management system for e-services that uses a client-aware model. Client-awareness is obtained by reproducing the behaviour of real clients when they give feedback and when they choose services. To this aim, we gather data from popular web sites specialised in client reviews. From the analysis of such data, we derive a probabilistic client model to obtain feedback and make choices according to a service category and a client typology. Moreover, by combining together different kind of services, we form a list of composite services (also called *packets* throughout the paper) sorted by a single reputation value. We evaluate the soundness of our reputation system through a simulator that mimics the reputation management in a scenario in which a broker provides packets to different kind of clients. We also consider a client that intentionally provides false feedback.

The paper is organized as follows. Section 2 recalls related work and discusses similarities and contrasts with our proposal. In Section 3, we describe the reference scenario. Section 4 presents our procedure to compute the reputation of services. Section 5 validates the reputation management procedure. In particular, Subsection 5.1 shows how we derive a probabilistic model both for the client choice and the client feedback. In Subsection 5.2, we present a number of evaluations we have carried out. Finally, Section 6 concludes the paper.

2 Related Work

In the literature, there are proposals for reputation and trust management mechanisms in a service-based scenario. Here, we describe some relevant work in the area, and we compare our framework with the existing research.

Zhao et al. [5] consider the experience of real users who give feedback about e-services. Consumers can obtain the set of past feedback that matches best with their context. This set is selected using a function that expresses the distance between the current context and past contexts. Also, feedback obtained are affected by a fading factor, which is used to weight less old feedback. However, a specific formula for expressing the trustworthiness of services is not given. The framework has been tested with real users in order to give findings in terms of accuracy about feedback proposed to a user. Simulator/software are not publicly available. In fact, this is left as future work.

Multi-dimensional trust for a grid scenario is proposed in [6]. Here, reputation values derive from various parameters, *e.g.*, quality, success, and others. A software agent preliminarily computes single reputation values related to each parameter. Then, a global reputation value is obtained by means of a formula that gives weights to each of the parameters. The paper does not describe how services are chosen and evaluated.

The framework described in [7] dynamically computes the reputation of a service provider as specified in a Service Level Agreement (SLA). A SLA is a legal contract between a provider and a consumer, specifying bilateral agreements regulating business relationships. Evaluations of the robustness of the proposed reputation mechanisms are carried out considering possible misbehavior of the service provider.

The proposal in [8] investigates how reputation is propagated in composite services, from the composition to the component services. The method enables the composite service to provide a fair distribution of reputation values so that a component service is neither penalized nor awarded for the bad and good performances, respectively, of other component services.

Coetzee and Eloff in [9] consider reputation issues in a client-broker-service provider scenario. The broker forwards the client requests to the service provider. The work is focused on the broker, and proposes a system based on past interactions to increase or decrease the trustworthiness of this third party. The formula through which the reputation is computed exploits fuzzy logics. The reputation depends on multiple parameters, like the skills provided by the broker, and how consistent the broker actions are.

The proposal in [10] deals with a formal model for a reputation management in composite services. Each composite service is linked to a monitoring agent that raises exceptions when something unexpected happens. The single component services are not fully described in the formalization. There are hints to a formal analysis that could be carried out, but no details are given.

Work in [11] presents a decentralised model to store trust values about websites. A plug-in for a web browser has been developed, to show the reputation of a website, according to past experience of users.

Some work deals with the design of ontologies to refine the client's request by defining specific requirements. For example, the ontology in [12] specifies cryptographic primitives that must be used in a communication to protect the service, and the features of the Certification Authority that guarantees for that service. That work is thus oriented to cryptographically safeguard the client. Reputation issues are not directly addressed. [13] focuses on the PeerTrust language, allowing a trust negotiation phase between the entities at stake. This work particularly aims at discovering if the clients act dishonestly in presenting faked credentials to support their requests.

Finally, it is worth noticing that reputation systems naturally suffer from the problem of reporting false feedback. This issue has been considered in depth in the area of Mobile Ad hoc Networks (*MANETs*) and Opportunistic Networks (*OppNets*) where collaboration among devices is fundamental, and malicious ones can provoke serious disservices to the network. To tackle this issue, several reputation mechanisms in the area exploit both direct and indirect observations [14,15,16,17]. In particular, these two parameters are often weighted to provide a different relevance in the calculus of the reputation of a device. In the area of service-oriented architectures, the issue of false reviews is analysed in [18], where the authors survey the trust and reputation mechanisms used by popular web-sites, such as Amazon, eBay, Advogato, Epinions, and others. Many of them adopt alert-based techniques to highlight unexpected reviews.

Although the literature has considered - and still focuses on - scenarios and issues similar to the ones we deal with in this paper, the starting hypotheses and the proposed solutions are different with respect to related work. To the best of our knowledge, the particular scenario in which a client requests a set of *packets* to a broker, and, based on a single reputation value, that client chooses among them the one that is closer to her typology, has not been considered in the literature. Despite the computation of the reputation recalls the previous proposals in the area of MANETs, *i.e.*, the use of weights to give more relevance either to past history or to new experience, our opinion is that this approach fits well in the area of e-services too. Also, an analysis on how clients choose services and give real feedback has not been considered in related work (at least, in the area of information technology). Moreover, a probabilistic model of clients has not been derived. Finally, it is worth noticing that some work consider as malicious the service providers and the broker. In this paper, we tackle the issue of malicious clients.

3 Reference Scenario

The reference scenario that we consider throughout the paper is illustrated in Fig. 1. We examine a system in which an online service broker B provides a (sorted) list of composite services CS to a client C. Each composite service is composed by component services S^i. Somewhere, we will refer to composite services as simply *packets*.

For instance, the client is a traveller willing to book a trip via B. The services requested by C are accommodation, transportation, and refreshment. For example, each packet CS^j could consist of: hotel S^j_H, car rental S^j_{CR}, and restaurant S^j_R [1].

Roughly, C asks B a list of composite services CS^i. B provides that list, ranked according to a real numeric value, ranged over $\{1.0, \ldots, 5.0\}$. C chooses a particular CS from that list. Then, C experiences the component services constituting CS. After completion of the trip, C assigns a value to each component service, reflecting her satisfactions. According to the collection of the single new values, B evaluates the current values of each component services and calculates a new ranking of composite services, to be shown to the next client.

The numerical values, according to which the packets are ranked, are the *reputation* values. We define the reputation of a (possibly composite) service as a real number expressing the *quality* of that service. This number is obtained through the collection

[1] Here, we simplify the scenario, by considering that all the possible accommodation services are represented by a hotel, all the possible transportation services are represented by a car rental, and all the possible refreshment services are represented by a restaurant.

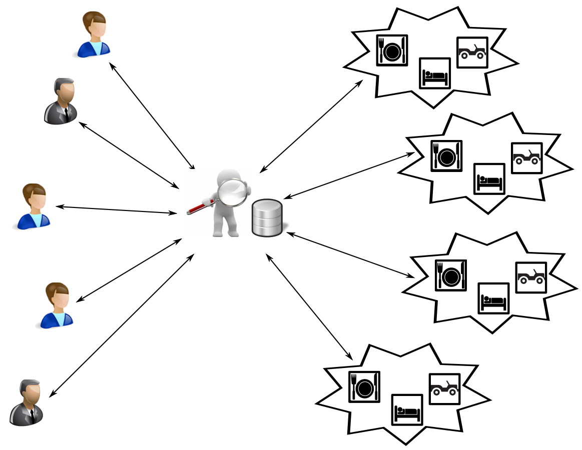


Fig. 1: Reference Scenario: Clients-Broker-Packets

of direct experiences of past clients. The quality of a service depends on a set of parameters characterizing that service. Reasonably, the parameters affecting the service quality could be, *e.g.*, the room price and cleanliness, the food quality, the comfort and punctuality of transportation, and, transversely, the staff kindness.

Choosing a service based on its reputation value has been recently considered in the literature. For example, Baldoni et al., in [7], consider a scenario where a provider supplies services to consumers. The provider-consumer relationships are regulated through so called Service Level Agreements (SLA), that specify a set of requirements for which consumers are willing to choose a service offered by a certain provider. The reputation of the service provider is specified in the SLA, both in terms of an initial reputation value, and of the functions used to compute and update such value.

Here, the scenario is different, since we deal with packets and an intermediate figure between the services and the client, *i.e.*, the online broker. However, we can assume that a SLA exists between B and each component service S^i, and that the initial reputation value of each S^i is specified in the related SLA.

In the following, we detail our reputation management protocol. In particular, we show how the reputation values associated with the services are bootstrapped and updated, in order to provide renovated rankings to clients.

4 Reputation System Architecture

The procedure for requesting and experiencing a composite service is the following:

1. The client asks the broker a packet (a hotel + a restaurant + a car rental).
2. B presents a list of CS, ordered by reputation values. The list is ordered from the packet with higher reputation to the one with lower reputation.
3. C chooses a packet according to her preference value (see Section 5.1), and experiences the chosen CS.

4. B sends C a questionnaire to evaluate C's experience with CS.
5. C fills in the questionnaire by giving feedback on the component services constituting CS, according to the model presented in Section 5.1.
6. The broker updates the reputation of the single services, and forms new packets and a new list of CS to be proposed to the next client.

We focus on step 6, *i.e.*, the computation and updating of the single service reputation. Then based on these new values, a new list of packets is made.

Once the client has expressed her feedback for a service, the broker updates the reputation value for that service. In line with related work in the area, we think that the new value should depend both i) on the feedback given by the last client and ii) on past experiences. The following formula generically indicates that the new reputation is a function f of the old reputation and the last feedback.

$$R^S_{new} = f(R^S_{fb}, R^S_{old})$$

In particular, we propose the next formula to compute the new service reputation:

$$R^S_{new} = R^S_{fb} * w_{fb} + R^S_{old} * w_{old} \tag{1}$$

where:

- R^S_{fb} denotes the feedback given by a client on the service S;
- R^S_{old} is the old reputation value that depends on clients past feedback;
- w_{fb}, w_{old} are weights ranging over $\{0, \ldots, 1\}$ and $w_{fb} + w_{old} = 1$.

The weights can be opportunely tuned in order to give more or less importance to history rather than to new feedback.

Finally, when a reputation value is assigned to each service by the broker, it creates a new updated list of packets. In our scenario, the broker starts considering the hotel, the restaurant, and the car rental with the highest values of reputation, and it puts them together in a single packet associating a unique value of reputation, see Fig.2. In particular, this value is the weighted mean of the single reputation values of the components. Subsequently, the broker selects the hotel, the restaurant, and the car rental with the second highest values of reputation, and it forms the second packet. The procedure is repeated until all services belong to a packet[2].

Although we give the same importance to each service in the weighted mean, we note that other mechanisms can be used to form and sort the packet list. For example, clients may give more relevance to hotels, rather than to restaurants and car rentals. According to this preference, the reputation value of the single service could be calculated giving a higher weight to the hotels in the weighted mean.

5 Validation

This section aims at validating the formula proposed in Section 4, by characterising in a specific way each actor involved in our scenario, *i.e.*, a set of clients, a broker, and a set of e-services. In particular:

[2] We consider the same number of hotels, restaurants, and car rentals

1°	H_8= 4.8	R_5= 4.6	C_3= 4.9	Rep_1= 4.8
2°	H_6= 4.7	R_3= 4.5	C_1= 4.7	Rep_2= 4.7
3°	H_2= 4.5	R_6= 4.3	C_8= 4.4	Rep_3= 4.4
4°	H_1= 4.3	R_4= 4.2	C_9= 4.2	Rep_4= 4.2
	• • • • •			
N°	H_5= 1.2	R_7= 1.5	C_3= 1.7	Rep_N= 1.4

Fig. 2: A list of packets sorted by reputation values

- The broker is an agent that interfaces services and clients, by following the protocol given in the previous section.
- The services are hotels, restaurants, and car rentals.
  - Each of them enters the system with an initial reputation value fixed in agreement with the broker. For example, this value may be fixed in a Service Level Agreement *(SLA)*. However, for our validation purposes, we decide to adopt completely random initial reputation values. We justify this choice to test the goodness of our proposal, in terms of proving: 1) if the reputation values come to results comparable to the reference values (see below); 2) how fast the reputation mechanism is in adjusting the initial values.
  - In order to validate our reputation system, we need reference values for the reputation of each service in a steady state. For each service, we take as the set of reputation reference values that one surrounding the category reported in the website. As an example, for us, a reference reputation value for a 5 star hotel ranges over $\{4.51 \ldots, 5\}$, and for a 4 star hotel over $\{3.51 \ldots, 4.5\}$.
- We consider three categories of clients: solo traveller C_{st}, family C_f, and businessman C_b. Following, we define how a client chooses a packet and how a client gives feedback regarding her experience.

5.1 Client model

Modeling a client choice As presented in Section 3, each client chooses a composite service CS from a list ordered by decreasing reputation values. What we propose is to base this choice on the numerical closeness to a preference value v associated to each client. This value represents the preference of a client.

The way through which we calculate the preference value v has been modeled by considering behaviors of real clients. In particular, we examine popular websites offering travel advices about hotels, restaurants, and car rentals[3].

Regarding hotels, we consider a subset of the 430 hotels in New York City reviewed on Tripadvisor.com. In particular, for each hotel class (from 5 to 1 star), we examine the

[3] All the surveys described in the paper refer to data gathered from websites in April, 2011.

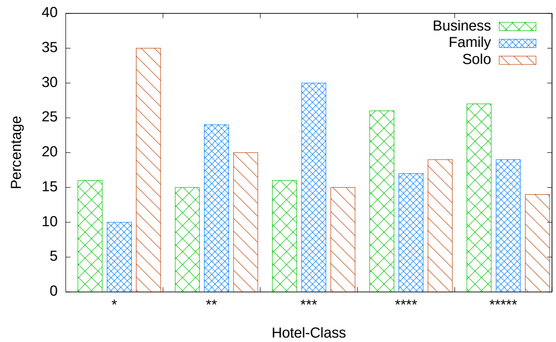


Fig. 3: NYC hotels: preference of clients. Percentage of clients choosing NYC hotels, per client typology and hotel class

first twenty hotels ranked by Tripadvisor, except that for the 1 star hotels, for which the website gives only a list of twelve hotels. For each hotel, we take the total number of clients C (businessmen C_b + solo travellers C_{st} + families C_f) reporting a review, and we calculate the ratio C_b over C, C_{st} over C, and C_f over C, representing, resp., the percentage of businessmen, solo, and families that have chosen that hotel on the totality of the three categories. Then, we calculate the mean value over all the percentages, per clients typology. Upon normalization, we obtain the results illustrated in Fig. 3. For example, we obtain that, on average, about 27% of businessmen prefer a 5 star hotel, 26% of them choose a 4 star hotel, 16% stay at a 3 star hotel, while 15% and 16% choose, respectively, a 2 star and 1 star hotel. Generalizing the results in Fig. 3, we assign a value v_h to each businessman, solo, and family traveller. This value is comprised between 1 and 5, and it is distributed as in the figure.

Regarding restaurants, we consider the almost 7000 restaurants in New York City revised on Tripadvisor. The website distinguishes them according to the price range, between \$ and \$\$\$\$. For each of these categories, we select a number of businessmen, solo, and families (with children) reviewers, and we compute the corresponding ratios. By following the same reasoning carried out for the hotels, we obtain the results illustrated in Fig. 4. As above, we assign a value v_r to each businessman, solo, and family traveller. This value is comprised between 1 and 5. Given that restaurant categories range from \$ to \$\$\$\$, we normalize them between 1 and 5.

Finally, we consider the website viewpoints.com, giving advice on best car rentals (www.viewpoints.com/Rental-Cars). The website distinguishes among budget-conscious, family, and business travellers. However, there is no notable difference between the ranking due to all the reviewers, and the ranking due to a particular category of reviewers. Thus, we decide to study the ranking due to all reviewers. We notice that the majority of the services have a similar number of reviews, meaning that they have been chosen with a similar frequency. Thus, we decide to model the choice of a car rental in a random, but uniformly distributed way. In particular, the car rental preference value v_{rc} is a random value uniformly distributed between 1 and 5.

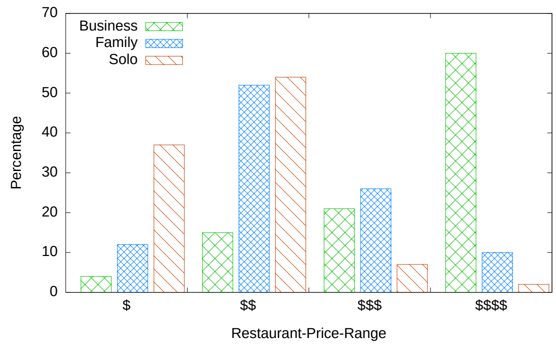


Fig. 4: NYC restaurants: preference of clients. Percentage of clients choosing NYC restaurants, per client typology and restaurant price range

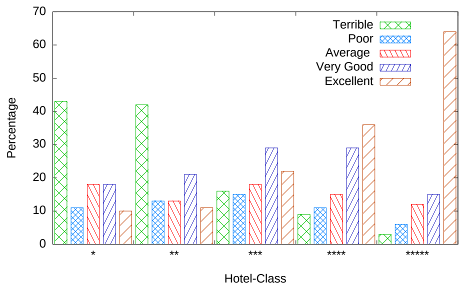
The preference value v associated to each client is obtained by computing the mean value between v_h, v_r, and v_{rc}. Thus, we consider a client giving the same importance to all the component services. However, it would be possible to give more relevance to one service rather than another one by computing a weighted mean.

Modeling a client feedback Once the client has experimented the composite service, the broker asks her to provide feedback. Here, we propose a probabilistic feedback model, based on real advices published on Tripadvisor.com. As for modeling a client's choice, we consider the New York City restaurants and hotels.

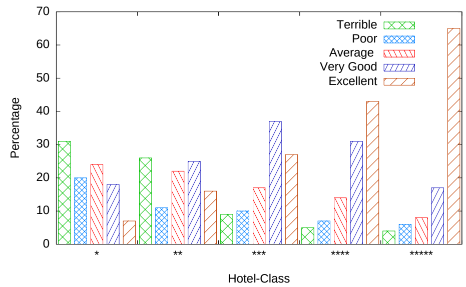
First, we describe our survey about hotels. We consider the Tripadvisor ranking, given per hotel class. For each class ranking, we select the hotels listed in the first four pages. Each hotel has a set of associated reviews. Each reviewer can judge a hotel with five marks: Excellent, Very good, Average, Poor, Terrible. It is possible to filter reviews per client typology, thus we select businessmen, families, and solo travellers reviews. Then, we calculate the percentage of businessmen, families, and solo giving a certain mark to hotels belonging to a certain class.

As an example, considering the NYC 5 star hotels, on the totality of 613 businessmen reporting reviews, 393 give an Excellent mark (64%), 92 businessmen a Very good mark (92%), 72 an Average mark (12%), 35 a Poor mark (6%), and finally 21 a Terrible mark (3%). The distribution of feedback, per client typology and hotel class, is illustrated in Fig. 5.

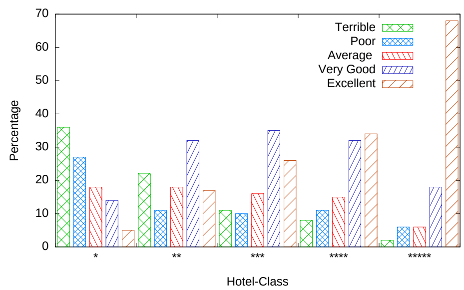
Tripadvisor does not allow to filter restaurant reviews according to the client's typology. Thus, we consider a generic traveller. For each restaurant category (price range: from $ to $$$$), we collect the reviews of the first four pages in the NYC restaurants ranking provided by Tripadvisor, and we calculate the percentages of Excellent, Very Good, Average, Poor, and Terrible marks. Results are illustrated in Fig. 6. As an example, we can see that 44% of clients consider a 4$ NY restaurant excellent, 33% give a very good mark, 17% think that $ NY restaurants are on average, and 4% and 2% are unsatisfied, giving, resp., Poor and Terrible marks.



(a) NY hotels: Business feedback



(b) NY hotels: Families feedback



(c) NY hotels: Solo travellers feedback

Fig. 5: NYC hotels: Clients' feedback. Percentage of clients giving a certain feedback, per client typology and hotel class

Finally, reviews about car rentals were not sufficient to derive a probabilistic distribution of clients' feedback. Thus, we decide to consider a uniform distribution of feedback, ranged over $\{1.0, \ldots, 5.0\}$.

Each service is associated to a default classification (*e.g.*, restaurants are classified by price range, and hotels are classified by stars). When a restaurant (respectively, a hotel) is evaluated, a client feedback is probabilistically obtained according to the percentages given in Fig. 6 (respectively, Fig. 5).

For example, a 4$ restaurant is judged *Excellent* with a probability of 44%, *Very good* with a probability of 33%, *Average* 17% and so on. Moreover, since we consider as reputation values real numbers ranged over $\{1.0, \ldots, 5.0\}$, the textual feedback is uniformly mapped to intervals as in Table 1.

Malicious clients Generally, online reputation systems suffer of the issue of malicious feedback. Indeed, it is possible to intentionally post false reviews. Actually, there exist mechanisms to detect if a given feedback is likely or not. For example, in the eBay system for e-commerce (www.ebay.com), when a feedback is different in respect to what a seller would have expected, eBay queries both the client and the seller, in order to investigate on that feedback. Other websites allow users to add a notice about a review, if they do not agree on it. In this paper, we acknowledge this weakness, and to

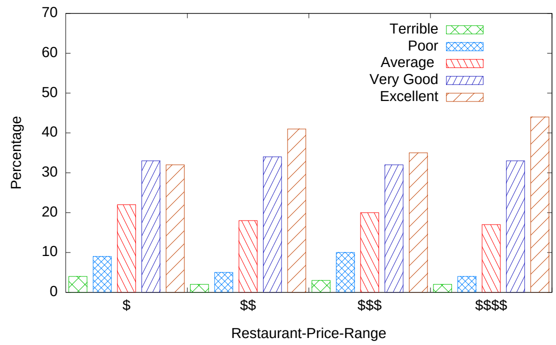


Fig. 6: NYC restaurants: Clients' feedback. Percentage of clients giving a certain feedback, per restaurant price range

Table 1

Mark	Feedback Values
Excellent	[4.51 , . . ., 5.0]
Very good	[3.51 , . . ., 4.5]
Average	[2.51 , . . ., 3.5]
Poor	[1.51 , . . ., 2.5]
Terrible	[1.0 , . . ., 1.5]

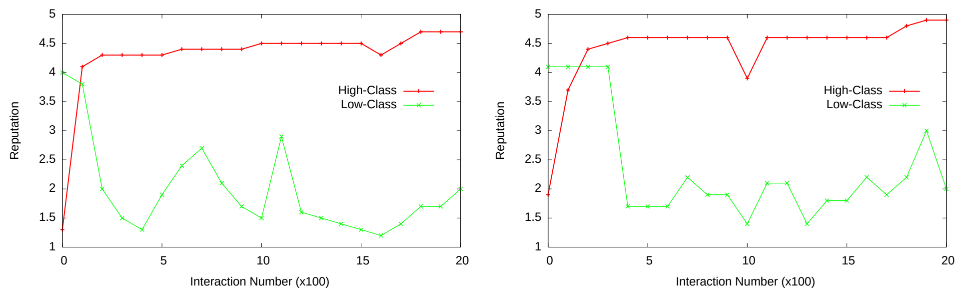
prove the robustness of our proposal, we consider a setting where not all the totality of clients are honest clients. For example, a dishonest client could be one who gives feedback in a completely random way. However, in our scenario, malicious clients are those whose feedback are mirror-like to those in Figures 5 and 6. According to the trend shown in the figures, an *Excellent* mark is given to a high-level service (*e.g.*, a 5 star hotel) with high probability. Thus, for us, a malicious client gives a *Poor* mark with that same probability.

5.2 Experimental Results

Here, we present some experimental results obtained through a study aiming at characterizing the behavior of the reputation management procedure shown in the previous section. The study is performed implementing an ad-hoc simulator that mimics our framework by letting: 1) the broker propose the current list of packets to the client; 2) the client randomly choose a packet following her preference value v; 3) the client randomly give feedback to each component according to her model; 4) the broker update reputation values of component services according to current feedback and old reputation following the formula in Section 4; and 5) the broker update the list of packets. A number of different interactions is realised in subsequent steps.

The simulator has been developed in JAVA (www.java.com) and can easily run on traditional laptops or desktop computers. Computational time is less than 8 sec. for two thousand interactions, with an Intel Core2 Duo 2.4Ghz. Each actor of Section 3 has been created as a single object. To run a simulation, it is enough to set the number of clients. The simulator is available online at http://www.iit.cnr.it/staff/gianpiero.costantino/CNR-PersonalPage/Simulator.html .

We ran several simulations with different values for w_{fb} and w_{new} (see expression 1) in Section 4. According to different weights, one can give more relevance to past feedback w_{fb} or to new feedback w_{new}.



(a) Hotel reputation trend (b) Restaurant reputation trend

Fig. 7: Reputation trend established for two distinct services

Honest Clients Figures 7a-7b show the trend of the reputation values in a setting where all clients are honest, *i.e.*, they provide honest feedback. In the simulations, we consider 2000 interactions, in each of them a client asks the broker the list of packets, she chooses one, experiences the component services, and reports a feedback for each of them.

In particular, the figures show the reputation trend of two hotels and two restaurants (for each pair, one of high class, the other one of low class) that was obtained with $w_{fb} = 0.4$ and $w_{old} = 0.6$. We recall that initial reputation values have been randomly given. According to our probabilistic client model, the selected services quite quickly obtain reputation values very close to the *reference values*. For example, reference values for high class hotels and restaurants are in $\{4.5, \ldots, 5\}$ In the figures we can see that the reputation quickly come to comparable values. In particular, services with high reference values present a flatter trend —cross-shaped line— with respect to the ones with lower reference values —x-shaped line—. This is justified by the client model. Indeed, the probability to have *Excellent* or *Very Good* marks is very high for *high-class* services, and, so, they are less subject to sudden changes in the reputation trend.

Malicious Clients Here, we aim at finding the optimal weights in Expression 1 in order to suffer as less as possible from malicious feedback. Thus, we ran several simulations, with different values for weights and different percentages of malicious clients. We consider those in Table 2.

Table 2: Honest/Dishonest Percentages of Clients

Honest	Dishonest
100	0
90	10
80	20
70	30
60	40
50	50

In Figure 8 we show the most relevant results we have obtained. The figure shows results for a 4 star hotel. On the left column, the reputation trend is shown in a setting with a low amount of malicious clients (up to the 30% of the totality), while in the right column a higher percentage is considered (up to 50%).

When we give more importance to new feedback with respect to older ones, the reputation trend is less stable. This is due to the fact that few new positive (resp., negative) feedback are sufficient for rapidly increasing (resp., decreasing) the service's reputation. Moreover, the interactions of past customers are quickly forgotten. Hence, an attacker may easily falsify the reputation of a service, see, *e.g.*, Fig. 8a. Also, Figure 8b shows how a relevant amount of dishonest clients can provoke a completely distorted reputation trend in services.

On the other hand, when using very low weights for new feedback (*e.g.*, $w_{fb} = 0.1, w_{old} = 0.9$), the resulting trend is flatter but this may be counterproductive since the reputation system may not be reactive enough. This case is shown in Figures 8c-8d.

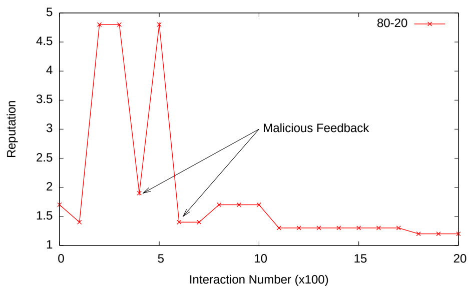
The best trade off between w_{fb} and w_{old} is presented in Figures 8e and 8f. A higher importance is given to old feedback. Nevertheless, new interactions are properly considered ($w_{fb} = 0.3$ and $w_{old} = 0.7$). Figure 8f highlights that these values of w_{fb} and w_{old} allow our system to be quite robust even in presence of a high percentage of malicious clients. Indeed, the resulting trend is not affected by substantial modifications.

6 Conclusions and Future Work

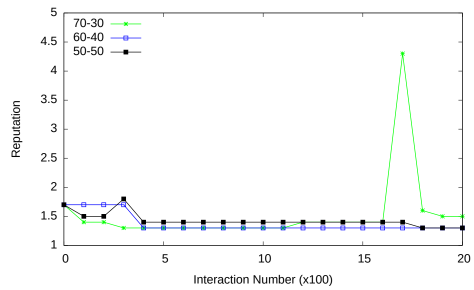
In this work, we have proposed a reputation system for different kind of services that are collected together in order to form packets. In particular, a client is willing to book a complete trip (composed of a hotel, a restaurant, and a car rental service), based on a reputation score provided by a broker.

Reputation systems for services are usually built on feedback expressing the satisfaction of past users. Before selecting a service, a user may consider its reputation value, derived by past feedback usually available on websites specialised in customers' reviews. Also, the selection depends on the user attitudes and preferences.

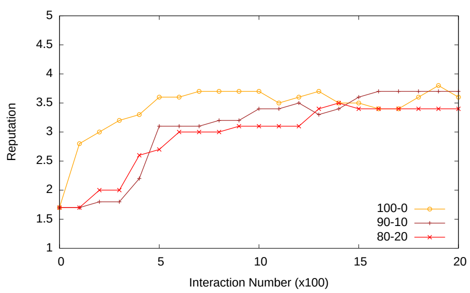
To model as close as possible the behaviour of users in giving feedback and making choices, we gathered data from ones of the most popular travellers reviews websites: *(www.tripadvisor.com)* and car rental site*(www.viewpoints.com)*. From the analysis of such data, we derive a probabilistic model for three kinds of clients: businessmen, families, and solo travellers. We do not experience with real clients choices and feedback.



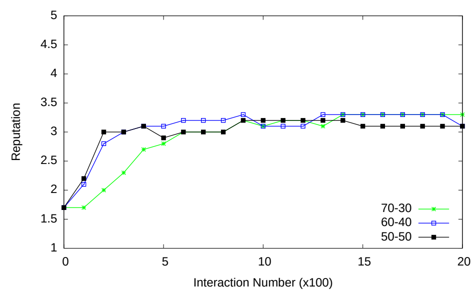
(a) Weights: $w_{fb} = 0.8$ and $w_{old} = 0.2$



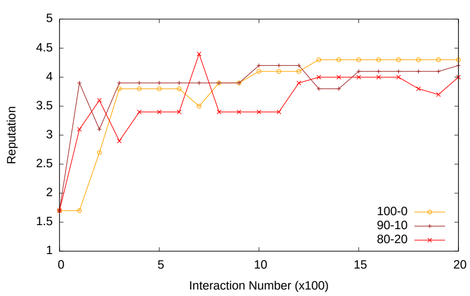
(b) Weights: $w_{fb} = 0.8$ and $w_{old} = 0.2$



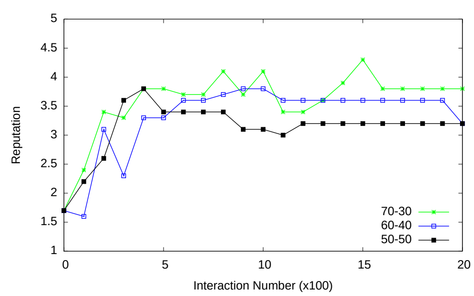
(c) Weights: $w_{fb} = 0.1$ and $w_{old} = 0.9$



(d) Weights: $w_{fb} = 0.1$ and $w_{old} = 0.9$



(e) Weights: $w_{fb} = 0.3$ and $w_{old} = 0.7$



(f) Weights: $w_{fb} = 0.3$ and $w_{old} = 0.7$

Fig. 8: Reputation trend varying w_{fb}, w_{old}, and percentage of dishonest clients

Nevertheless, the model we derived comes from real preferences and real reviews. The efficacy of the model has been evaluated by simulating a reputation system able to get, as input, feedback of past clients, and return a reputation value for each of the component service, and, hence, for the packet. Simulations show that our mechanism works well up to a certain number of malicious feedback.

We think that other interesting directions could be investigated. First, malicious feedback may lead to unreal reputation values. We aim at extending our work with a proactive component where alarms can be raised when something is suspected to go wrong. Secondly, assuming that an initial reputation value is fixed between a broker and a service in a business agreement, anomalies between that value and the value calculated according to our reputation system may lead to re-considering the agreement and may cause a change of the reputation values of the involved services and/or the broker. We leave this for future work based on contracts.

References

1. Erl, T.: Service-Oriented Architecture (SOA): Concepts, Technology, and Design. Prentice Hall (2005)
2. S., I.I.A.: Software as a Service: Strategic Backgrounder (2001)
3. Weiss, A.: Computing in the clouds. netWorker **11**(4) (2007) 16–25
4. Rokach, J.Z.: Unlocking the Smart Grid. Electricity Journal **23**(8) (2010) 63–71
5. Zhao, W., Sun, H., Huang, Z., Liu, X., Kang, X.: A user-oriented approach to assessing web service trustworthiness. In: ATC. Volume 6407 of LNCS., Springer (2010) 195–207
6. Griffiths, N.: Task delegation using experience-based multi-dimensional trust. AAMAS '05, ACM (2005) 489–496
7. Baldoni, R., Doria, L., Lodi, G., Querzoni, L.: Managing reputation in contract-based distributed systems. In: OTM. Volume 5870 of LNCS., Springer (2009) 760–772
8. Nepal, S., Malik, Z., Bouguettaya, A.: Reputation propagation in composite services. ICWS, IEEE (2009) 295–302
9. Coetzee, M., Eloff, J.: A framework for web services trust. In: Security and Privacy in Dynamic Environments. Volume 201 of IFIP. Springer (2006) 74–86
10. Zhengping, L., Xiaoli, L., Guoqing, W., Min, Y., Fan, Z.: A formal framework for trust management of service-oriented systems. In: SOCA, IEEE (2007) 241 – 248
11. Mahmood, O., Selvakennedy, S.: Modeling "web of trust" with web 2.0. Enformatika Journal **18** (2006) 122 – 127
12. Galizia, S., Gugliotta, A., Domingue, J.: A trust based methodology for web service selection. In: Semantic Computing. (2007) 193–200
13. Olmedilla, D., Lara, R., Polleres, A., Lausen, H.: Trust negotiation for semantic web services. In: Semantic Web Services and Web Process Composition, Springer (2004) 81 – 95
14. Liu, J., Issarny, V.: Enhanced reputation mechanism for mobile ad hoc networks. In: iTrust. (2004) 48–62
15. Michiardi, P., Molva, R.: Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In: CMS, Kluwer (2002) 107–121
16. Yan, Z.: A conceptual architecture of a trusted mobile environment. In: SecPerU, IEEE (2006) 75–81
17. Bella, G., Costantino, G., Riccobene, S.: Spreading of reputation values in a mobile network. In: SoftCOM, IEEE (2008)
18. Jøsang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. Decision Support Systems **43**(2) (2007) 618 – 644