View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Helsingin yliopiston digitaalinen arkisto

Reputation Management Survey

Sini Ruohomaa, Lea Kutvonen
Department of Computer Science
University of Helsinki
{lea.kutvonen,sini.ruohomaa}@cs.helsinki.fi

Eleni Koutrouli
Department of Informatics and Communications
National University of Athens
ekou@di.uoa.gr

Abstract

Electronic markets, distributed peer-to-peer applications and other forms of online collaboration are all based on mutual trust, which enables transacting peers to overcome the uncertainty and risk inherent in the environment. Reputation systems provide essential input for computational trust as predictions on future behaviour based on the past actions of a peer. In order to analyze the maturity of current reputation systems, we compare eleven reputation systems within a taxonomy of the credibility aspects of a reputation system. The taxonomy covers three topics: 1) the creation and content of a recommendation, 2) the selection and use of recommenders, and 3) the interpretation and reasoning applied to the gathered information. Although we find it possible to form a trusted reputation management network over an open network environment, there are still many regulatory and technical obstacles to address. This survey reveals various good mechanisms and methods used, but the area still requires both a) formation of standard mechanisms and metrics for reputation system collaboration and b) standard metainformation of right granularity for evaluating the credibility of reputation information provided.

1. Introduction

Electronic markets, distributed peer-to-peer applications and other forms of online collaboration are all based on mutual trust, which enables transacting peers to overcome the uncertainty and risk inherent in the environment. Reputation systems provide essential input for computational trust as predictions on future behaviour based on a peer's past actions. Information about these actions can also be received from other members of a reputation network who have transacted with the peer. However, the credibility of this third-party information must be critically assessed.

The underlying goal in all reputation systems is predicting a peer's future actions, given knowledge about its past behaviour. This knowledge is ideally gathered via first-hand transactions, but it becomes costly to have to interact with every peer, including the malicious ones. To alleviate the cost, peers share their experiences through the reputation system, which makes it possible for the entire community to detect and isolate misbehaving peers effectively. We refer to this shared information as recommendations.

As no actors in the network are fully trusted, recommendation credibility must be critically assessed. Reputation systems have been widely used for various applications, and work on credibility assessment of the provided reputation information is becoming increasingly important as the technology extends to areas where more resources and business value depend on making correct trust decisions.

A peer can provide inaccurate reputation information for a variety of reasons. A colluding peer claims without grounds to have had positive experiences with another peer to increase its reputation, expecting some of its colluder's profits from defecting. A peer can also defame another peer by providing undeserved bad experience information, in order to increase its own reputation in relation to the target. Providing random information may allow a peer to benefit from participating in the reputation network without the cost of performing transactions itself. Furthermore, human users have social reasons for skewing their feedback. Finally, the targets of reputation information can change their behaviour over time, or discriminate against particular peers while cooperating with others.

All these kinds of misbehaviour in a reputation system make it important to separate accurate information from inaccurate. A thorough credibility analysis increases the system's resistance against malicious peers, if supported in the system's information and operation models.

We believe it is possible to create a trusted reputation management network to support open, online collaborations (e.g. [8]), and therefore analyze the maturity of current reputation systems. We compare eleven systems in terms of 1) the creation and content of a recommendation, 2) the selection and use of recommenders, and 3) the interpretation and reasoning applied to the gathered information. In the comparison taxonomy, the differentiating aspect proved to be the way credibility of reputation information was managed.

The rest of the paper is organized as follows. Section 2 describes our taxonomy, while Section 3 presents the discussed reputation systems. The following three sections present the analysis based on a division to three criteria sets: Section 4 compares approaches to recommendation creation and content. Section 5 analyzes the selection and use of recommenders. Section 6 discusses the reasoning and interpretation the systems apply to the gathered recommendations. We conclude by summarizing the findings and with recommendations for directions of further work on the area.

2. Taxonomy of credibility aspects

Our taxonomy of the credibility aspects of a reputation system covers three topics: 1) The creation and content of a recommendation, 2) the selection and use of recommenders, and 3) the interpretation and reasoning applied to the gathered information. The first topic determines whether the information in a recommendation provides sufficient basis for credibility analysis, be it implemented or left as potential extensions. The second and the third topics determine the quality of and the reasoning performed on the gathered information respectively.

The creation of a recommendation involves combining local experience information into a standard form to present to another actor. The encoded information can include both positive and negative experiences or focus on either, and reports can contain either single experiences or more general, aggregated opinions. The recommendation's target may be given a chance to comment on the statement as well.

The content of the final recommendation, both that of the rating or opinion and of other related information, determines the transparency of the creation and transfer process to the target. For separating new and fresh information from old, its time of creation or possibly the time of the transaction the recommendation relates to may be reported. Similarly, the recommendation can include a way to express confidence, or lack thereof, of the recommender in its statement. Finally, for systems where the recommendation is passed through mediators, additional information is needed to ensure the transparency of the mediation process.

A node gathering recommendation information can either actively select its recommenders from the set of all possible nodes, or simply allow and expect any knowledgeable nodes to provide recommendations. In addition, the gathering node can be the trustor itself or e.g. a centralized entity collecting all recommendations. These two dimensions form four different variants of reputation system operational modes [2], each with their own credibility qualities, threats and bias possibilities. If information is not gathered by the trustor itself, the transparency provided by the content of the recommendation becomes even more important. A central criterion to consider in all cases is the credibility of the recommender, which can affect both its selection and later the weight given to its recommendations.

Mediating recommendations brings its own threats and possibilities. While using mediators adds the threat of modifying and withholding information, it also provides a possibility to include additional information such as a meta-recommendation about the mediated information.

Once the recommendations have been acquired, they must be combined with any local experience and aggregated into a suitable format that could support a trust decision. There are several considerations that can be applied to the gathered data. The trustor or the information gatherer can analyze various background information to set the recommendations in a context. The trustor's own transactions with the trustee form the core of local experience, and to evaluate the recommender, the gatherer can consider both the past transactions with the recommender and the past recommendations received by it. Track record information can also be weighted based on recency.

Finally, the system may analyze the overall credibility of the processed information. A low amount of gathered information can be one source of uncertainty, and the quality of the sources has a strong effect on the credibility of the result. The general environment the reputation system operates in can be taken into account as well. For example, in some environments the fear of reprisals or a general tendency to reciprocate both in the positive and negative can skew the results.

3. Compared reputation systems

We have chosen eleven systems for our analysis: **eBay** [4], **Unitec** [7], **FuzzyTrust** [15], **REGRET** [14], **NICE** [9], **Managing the Dynamic Nature of Trust** (**MDNT**) [16], **PeerTrust** [17], **Managing Trust** [1], **Maximum Likelihood Estimation of Peers' Performance** (**MLE**) [3], **EigenTrust** [6] and **Travos** [11]. The systems represent a wide range of applications with different requirements for reputation analysis. If the reputation estimation should support e.g. a low-risk selection of an agent to download a given file from, the situation does not warrant as careful analysis as e.g. finding a business partner would. Current systems represent relatively low-risk environments, while reputation systems designed for more complex, business-to-business collaboration environments are also on the rise [10, 13].

Our main system selection criterion has been that it provides some kind of systematic description of the facilities available for credibility analysis. Credibility analysis adds complexity, and many reputation system descriptions have left it outside their scope or only suggested where to begin,

Table 1. Analyzed reputation systems

System	Application Area	Actors	Reputation Value
eBay	electronic marketplace	human	statistics, text
Unitec	generic framework	human	configurable
FuzzyTrust	multi-agent marketplace	agents	numeric
REGRET	multi-agent marketplace	agents	numeric, categorized
NICE	cooperative applications	agents	configurable
MDNT	online communities	agents	numeric, {0,..,6}
PeerTrust	online communities	agents	numeric, [0,1]
Managing Trust	online communities	agents	numeric, -1, 0, 1
MLE	online communities	agents	probability
EigenTrust	file sharing	agents	probability
Travos	Grid services	agents	probability

e.g. by treating recommendations as a service with its own trust category. Table 1 summarizes the systems selected.

The commercial **eBay** reputation system stores users' ratings linked to their profiles and related transactions, but leaves the credibility analysis to the human user. **Unitec** is also clearly directed towards a human user, but it performs automated credibility analysis as well. The content of a recommendation is not fixed, and the system could handle product recommendations with the same algorithm as well.

FuzzyTrust and **REGRET** are both designed for multi-agent marketplaces, but they apply rather different approaches to reputation estimation. In **FuzzyTrust**, local trust scores are generated through fuzzy inference, and aggregated to global reputation values. In **REGRET**, three viewpoints are applied as needed to infer a local reputation view, based on the social relations between peers. **NICE** is designed for cooperative applications on the Internet. Trustors are given signed receipts of successful transactions, "cookies", as a sign of some trust. These can be used to link actors into a weighted trust chain.

MDNT, **PeerTrust**, **Managing Trust** and **MLE** are designed for peer-to-peer community environments, which encompass both multi-agent marketplaces and cooperative applications and can host a multitude of activities: distributed applications, information exchange such as file transfers, and transactions as on online marketplaces. Their approaches to reputation are varying as well. In **MDNT**, a reputation estimate involves predicting the trustee's behaviour probabilistically, based on experience from a specific time period. **MLE** also uses a probabilistic approach, and considers the probability of recommenders to provide incorrect information. **PeerTrust** considers transaction and community contexts when estimating reputation. Finally, **Managing Trust** considers only negative experiences, and allows recommenders to remain anonymous.

EigenTrust is a reputation system for peer-to-peer file sharing. It relies specifically on a global, shared view of reputation. Credibility analysis is used in calculating each global reputation value, however. **Travos** aims to ensure good interactions between self-interested software agents in large scale open systems, such as the Grid. The agents provide interchangeable services, and reputation information is used to choose the most trustworthy partner. It bases reputation expression on Beta probability distributions.

4. Recommendation creation and content

We first compare different approaches to creating a recommendation, and what kind of experience is conveyed in it. The second part of the section focuses on what the content of a recommendation is, including how the conveyed experience is expressed and what metadata is provided to support the analysis. Table 2 summarizes the analysis.

When *creating a recommendation*, the recommender analyzes the experience information it has and produces a statement in an agreed-upon format to convey it to another node. Negative and positive experience information are both used in all systems except **Managing Trust**, where only negative experiences are stored. Including both allows the system to tell apart lack of experience and good experiences only, which is a valuable distinction.

A recommendation can be a single rating created after a transaction with the trustee in order to evaluate its quality or an opinion about the trustee formed by the aggregation of individual ratings. Both approaches seem popular. Aggregation saves traffic, providing better scalability, but reducing transparency.

If aggregated opinions are used for recommendations, the aggregation method used by the recommender becomes relevant. Of the 6 systems using opinions, **NICE** and **Unitec** do not specify the aggregation method and simply assume a policy is in place for determining it. The aggregation policy can be shared among all peers in a network, or locally defined by each peer. Not tying the system to a

Table 2. Recommendation creation and content

System	Rating / opinion	Opinion aggregation	Value	Time	Confidence	Other
eBay	rating		-1 / 0 / 1	T, R		free text
Unitec	opinion	unspecified	anything	R	√	recommender signature
FuzzyTrust	rating		$x \in [0, 1]$	T		
REGRET	opinion	weighted avg.	$x \in [-1, 1]$	(R)	√	
NICE	opinion	unspecified	$x \in [0, 1]$	R		recommender signature
MDNT	opinion	CCCI metric	$n \in \{0, .., 6\}$	R		
PeerTrust	rating		$x \in [0, 1]$	T, R		context, recommender signature
Managing Trust	rating		existence			mediator id
MLE	rating		0 / 1	T, R		
EigenTrust	opinion	exp. difference	$norm(p - n)$	(R)		
Travos	opinion	exp. counters	$p, n \in \mathbf{N}$	(R)	(√)	

single policy provides generality, allowing the system to be used with a broader application range. However, the user base within each application may be limited by the lack of a global agreement: subnetworks with separate policies are technically interoperable, but not necessarily semantically.

In **REGRET**, an opinion is calculated as the weighted average of single experience ratings, with more weight given to newer experiences. **Travos** and **EigenTrust** keep counters of positive and negative experiences, and use them to produce the aggregated opinion. **Travos** simply presents both counters, while **EigenTrust** calculates their difference and normalizes the value between the range $[0, 1]$. In both systems, a good experience is defined as the trustee providing the service promised, and there are no grey areas for single experiences. In **MDNT**, the opinion presented is calculated the same way as a trustor calculates a reputation value when it has local experiences, via the CCCI metric [5]. The metric considers the fulfillment of different expectations or criteria the trustor has for an interaction, and assigns them weights based on their importance and whether they have been clearly conveyed to the trustee.

Allowing the target to comment on a recommendation is very rare in reputation systems, with only one system supporting it. The **eBay** reputation system resides on a centralized server. It shows all ratings as a part of a user's profile, and the user can add a brief comment to each rating. In decentralized reputation systems, an actor may never know what kind of statements are made of its behaviour. The target's comments are also more challenging to include in an automated credibility analysis; **eBay** delegates the credibility analysis to the human user, who can to a degree evaluate the response based on its arguments.

For most systems, the main *content of a recommendation* is a single numeric value, but the scales used vary between discrete values and real numbers in a range. The **Unitec** recommendation is a generic wrapper around any form of value, and data aggregation is based on a local policy that fits the format. In **Managing Trust**, only the existence of a negative recommendation matters, and the value of a recommendation is not discussed nor used in the analysis.

In addition to the value, the recommendation contains various metadata items. All systems either include the target identity—a pseudonym or a stronger identifier—in the recommendation or provide it implicitly, e.g. as an answer to a query about a specific trustee. **Managing Trust** is the only system in our analysis to allow the sources of complaints to remain anonymous, but even this system uses the identity of the mediator storing them. Complete anonymity makes it impossible to estimate credibility based on the source of information. However, should the environment demand it, methods such as trusted third party mediators can be used to provide good privacy to recommenders.

Some metadata items provide support for giving varying weights to recommendations, for example to prefer recent information over old. Timing information is relevant for both the transaction a rating can be based on (indicated by T in Table 2) and the time when the recommendation itself is created (indicated by R). Transaction time information in the case of opinion-based reputation systems can be used to weight the single ratings in their aggregation. Recommendation time information is provided implicitly when a recommendation is generated as a response to a query (indicated by (R)), as is the case with **EigenTrust**, **REGRET** and **Travos**, and all recommendations become equal in that aspect. Not all systems use the time information they gather; e.g. **MLE** suggests a time stamp to be included in

Table 3. Selection and use of recommenders

System	Selection	Basis for Selection	Gathering	Mediation	Recommender Credibility
eBay	all		centralized		indirectly
Unitec	possible bias	user defined	trustor	√	recommendation record
FuzzyTrust	some bias	honesty, not loaded	trustor		transaction record
REGRET	no bias	social relations	trustor		trans. record+social relations
NICE	no bias	strongest trust path	trustee	√	transaction record
MDNT	some bias	credibility score	trustor		recommendation record
PeerTrust	all		trustor	√	trans. record/similarity
Managing Trust	all		trustor	√	trans. record for mediator only
MLE	undefined	subset of experienced	trustor	√	probability to lie
EigenTrust	all		a few peers		transaction record
Travos	all		trustor		recommendation record

the recommendation as a part of the key for storing it, not for weighting purposes.

The recommender's confidence on the information it provides is another weighing factor. None of the systems sharing ratings use it, as a good experience is considered well-defined and detectable. For opinions, confidence is related to the amount of information contained in the opinion and could be given subjectively, e.g. in **REGRET** and **Unitec**, where a recommendation is provided along with the subjective confidence of the recommender, or objectively, e.g. in **Travos**, which includes a confidence measure implicitly via the numbers of good and bad experiences.

Five of the analyzed systems use mediators: **Unitec**, **NICE**, **PeerTrust**, **Managing Trust** and **MLE**. Of these, all but **Managing Trust** and **MLE** include the recommender's digital signature in the recommendation to guarantee its integrity. In addition, **Unitec** mediators provide meta-recommendations. In **Managing Trust**, recommenders remain anonymous, but the credibility of the mediators storing the recommendations is estimated based on their transaction history.

5. Selection and use of recommenders

The criteria for recommender selection and use examine the sources and transfer of third party information used in the reputation system. The methods for selecting recommenders and possible weighting of the information they present have a strong impact on the quality of the resulting reputation estimate. If a group of faulty recommenders can block out other nodes from providing information to the trustor, there is little rigorous analysis can do. The same applies if the selection method simply does not reach potential recommenders who have experiences to share. In a general peer-to-peer network, the reputation system must be prepared to analyze a mix of correct and faulty information and have some ability to categorize the provided information accordingly.

In a peer-to-peer network with new nodes constantly appearing and old ones leaving, most nodes tend to only be connected to a few other nodes, while a handful of nodes become highly connected [12]. This phenomenon affects the amount of reputation information available about the nodes. A special group of actors to consider are the newcomers, who generally need a chance to prove themselves. While networks with low-cost identities must also protect against malicious identity changers, it is important to keep a balance, and not raise the threshold for joining the network as a newcomer too high. The recommender selection and use comparison is summarized in Table 3.

The selection of recommenders involves a trade-off between scalability and consistently gathering a good representation of recommenders. We have estimated whether the selection method can skew the result some way when compared to using all possible recommenders with the algorithm. However, as long as recommending is voluntary, there is a natural skew in the information to begin with.

Five systems use recommendations from all peers: **eBay**, **PeerTrust**, **Managing Trust**, **EigenTrust** and **Travos**. Of the remaining six, we estimate two to have basically no skewing caused by limiting the number of recommenders. **REGRET** groups peers according to their social context, such as their relationship with the trustee, and then chooses the most representative member of each group to give recommendations by fuzzy rules [14]. The method makes actual skews difficult to estimate, however, and open to interpretation. In **NICE**, the trustee tries to locate chains of first-hand recommendations from the trustor to itself.

In **FuzzyTrust**, recommenders are chosen through global weighting based on their number of performed transactions and local trust score. The weight is then compared to a threshold that is set based on the peer's role in the network—a superpeer with a high number of transactions has a higher threshold than a regular peer, for load balancing reasons. The influence of peers with many transactions is thereby somewhat reduced. In **MDNT**, recommenders are selected based on their credibility. This lowers the likelihood of new recommenders to get their voice heard, as peers who gain credibility fastest at first will continue to remain the most credible unless they get caught with false statements. **Unitec** may have a bias in the selection method, as it is left for the user to decide. For **MLE**, reputation estimation is based only on a subset of the existing recommendations, but the the selection method is not defined.

The most common solution for recommendation gathering in distributed systems is to leave it to the trustor. In **eBay**, recommendations are stored and analyzed on a centralized server and completed reports are served to trustors. Decentralized systems provide two exceptions to the norm as well. In **NICE**, the trustee itself stores signed "cookies", receipts that the signer has had positive experiences with it. It also gathers recommendations in the form of trust chains when needed. Information about negative experiences is stored on the trustor, as the trustee would have a strong incentive to omit it. In **EigenTrust**, reputation values are global. They are based on a majority vote of a few randomly chosen peers who first gather recommendations from the network, then weight them according to their trust in the recommender. Some steps are also taken to make the calculating peer unaware of the identity of the peer it is calculating a reputation value for.

The systems *using mediators* are marked in the fifth column in Table 3. As mentioned in the previous section, three of them use digital signatures to ensure that the recommendation is not modified on the way, but none have taken measures to defend against a mediator omitting information. In **NICE**, the effects of omission are borne by the trustee alone, as it must locate a chain of trust from the trustor to itself. Negative experiences are remembered by the trustor. In **Managing Trust**, the trustor does not know the original recommender, so it must evaluate, based on transaction experience alone, whether the mediator might have created negative recommendations of its own or omitted them. **MLE** does not consider mediator credibility. In addition, the trustee itself stores its recommendations, which raises a considerable credibility issue.

In **PeerTrust**, recommendations are saved distributedly, searchable by the trustee's id. The mediator does nothing but repeating the set of signed recommendations it is responsible of storing, and can omit information at will.

Unitec appears to realize the most benefit from mediators, through meta-recommendations. Each mediator in the path the recommendation takes can comment on how trustworthy it considers the statement from the previous actor. Recommendation queries are forwarded systematically to specific recommender groups, so mediators will typically have an opinion of the recommender they comment on.

Estimating the *credibility of the recommender* separately protects against peers that perform well in their transactions, but provide faulty recommendations. **FuzzyTrust**, **NICE**, and **EigenTrust** follow the simple model and assume a straightforward connection between transaction and recommendation trustworthiness. **REGRET** and **PeerTrust** follow the trend, although **REGRET** also incorporates social relationships, and **PeerTrust** suggests calculating a personal similarity measure as an alternative for estimating recommender credibility. Similarity measures are commonly used in product recommender systems, where credibility cannot be measured objectively.

The **eBay** reputation system supports only transaction trustworthiness directly, although a human user can also utilize some indirect information on the recommender, such as tone and form of ratings the recommender has given in general as well as the targets' comments to them.

Unitec, **MDNT**, **Travos** and **MLE** adjust the weight of the recommendation based on a specialized measure for how often the recommender has been correct or not in its past recommendations to the trustor. In **Unitec**, correctness is determined by the human user, while for **Travos** it is apparently automated. **MDNT** measures the distance of the recommendation from the actual evaluation of the trustee in the context of a transaction by the CCCI metric [5], to form its credibility measure. **MLE** calculates the probability of the recommender to lie based on its earlier interactions with the recommender. This can be done either separately for each peer or at the level of the whole network, in which case all peers have the same probability to lie.

6. Reasoning and interpretation

Once reputation information is collected, it must be aggregated into a suitable format to support trust decisions. The third set of criteria evaluate the analysis a reputation system applies in the gathered information. We examine how recommendations are aggregated into a reputation estimate, and what the interpretation given to the final estimate is. We also analyze whether the result is evaluated for internal credibility, or degree of certainty. Table 4 summarizes these three points of the analysis. We also look into the kind of information history stored and used and into how systems are adjusted to their application area and social setting.

The *recommendation aggregation* method is generally based on summing up the given recommendation values, multiplied by weights based on factors such as recommendation credibility. In **NICE**, the final reputation value is determined as the strength of the strongest chain, or the

Table 4. Reasoning and interpretation applied to the gathered information

System	Aggregation	Interpretation	Evaluation
eBay	Statistics and full data	report only	various data
Unitec	Weighted statistics	report only	unspecified
FuzzyTrust	Weighted average	threshold or rank	–
REGRET	Statistics, fuzzy inference	threshold or rank	per dimension
NICE	Strongest path's strength	threshold	–
MDNT	Weighted average	threshold	–
PeerTrust	Weighted average + context	threshold or rank	unspecified
Managing Trust	Weighted sum	threshold	iterative mediator evaluation
MLE	Probabilistic	threshold	–
EigenTrust	Weighted average	probability	–
Travos	Probabilistic, fit beta distribution	rank	–

sum of the strongest disjoint chains. The strength of a trust chain is equal to the lowest recommendation value between two nodes on it. **PeerTrust** uses transaction context in the weighting whereas a community context factor can be used to adjust the average result. In **REGRET**, reputation is calculated as the weighted sum of a set of reputation values mapped to the local experience, the social relationships of the trustee, and the correlation between reputation categories. The weights are reliability measures estimated for each reputation value.

Travos uses a probabilistic aggregation that fits the experience information gathered into beta probability distributions, and aggregates different items by summing the distributions. For each item, the probability distribution is adjusted towards an uniform probability based on how likely it is that the recommender is correct. In **MLE**, a reputation value is the probability of a peer to cooperate in a transaction. It is chosen to maximize the probability of the observed available experience information in the predictive model, which also takes into account the calculated probabilities of recommenders to lie.

The final *interpretation of the result* is most commonly threshold-based: if the trustee's reputation value is high enough, the trustor will decide to transact with it. This approach is taken by **NICE**, **MDNT**, **Managing Trust** and **MLE**. On the other hand, **FuzzyTrust**, **REGRET** and **PeerTrust** do not specify how the final measure is interpreted, but both threshold- and rank-based approaches are possible. Only **Travos** explicitly uses a rank-based approach, where a trustor orders a group of potential partners based on their reputation for selection purposes. **EigenTrust** uses the resulting reputation value as a probability of selecting the trustee as the peer to transact with. Finally, **eBay** and **Unitec** do not perform the interpretation at all, as they present a report to a human user. In **eBay**, both interpretations are possible, with thresholds more common. Generally, rank-based selection is only usable when several potential partners can provide a similar service and are therefore interchangeable.

Evaluating the result in terms of credibility or reliability is done only rarely. **REGRET** calculates a reliability value for each type of reputation, based on a variety of factors such as the available information, the variability of the individual ratings, the confidence of the recommender and the social relationships. It also estimates a reliability measure for the final estimated reputation. Both **eBay** and **Unitec** provide reports, for which varying metadata to support an evaluation is available. However, as **Unitec** does not define the format of the report in the system, the availability of this metadata to the user depends on the system configuration. **PeerTrust** also proposes the compilation of a confidence value associated with the reputation estimate, but instead of a specific measure, it presents a tentative suggestion for the value to reflect the number of transactions and the standard deviation of the recommendations depending on different communities.

The credibility of the estimated reputation value depends also on the following factors: the history of transactional information, as the more information is used the more accurate the result will be; and on giving higher weight to recent information about the peer's behaviour, as it is more likely to be indicative of its future behaviour.

As noted earlier, the information history available to a reputation network can be divided into two categories: transactional history, and the history of accurate recommendations. The full transaction history is stored by **eBay**, **PeerTrust** and **Managing Trust**, while **MLE** uses a subset only. In **NICE**, the trustor and the trustee divide the task of storing the transactional history between them, as recommendations which expire after a given period of time. **REGRET**, **Travos**, **EigenTrust**, **MDNT** and **FuzzyTrust** use both information about transactions and the recommender's track record as a recommender. **Unitec** does not store history, but a single current opinion value that is updated by each experience item. This is a scalability trade-off in exchange for some lost details.

Recent transactions are given more weight in reputation estimation in many systems, which is the usual reason to include time information as summarized in Table 2 earlier. **FuzzyTrust** uses transaction times along with other factors to estimate weights for recommendations when calculating the reputation view. In one proposed version of reputation estimation metric for **PeerTrust**, transaction times are used as a part of a context factor that weighs recommendations. Both **REGRET** and **MDNT** use transaction times when calculating an opinion-based recommendation, but do not share them. In addition, **MDNT** uses recommendation time information for giving weight to recency in recommender credibility updates. While **eBay** does not use time information in the numeric data, it sorts the text comments by recency and makes the times of transactions and recommendations visible to the user. **NICE** uses the recommendation time to prune old recommendations.

Unitec does not specify its aggregation method nor its credibility calculation, but since it stores the time of recommendation, it could be used in an analysis. **Managing Trust**, **MLE**, **EigenTrust** and **Travos** do not use transaction or recommendation recency information, considering all experience information equal in that sense.

We analyzed whether systems were *adjusted to their environment of operation*, and found two design approaches concerning this: 1) Allowing more than one level of reputation calculation is an adjustment to the technical network environment. It allows systems to differentiate between very clear or "routine" decisions and situations requiring more careful analysis. 2) Addressing the tendency for reciprocity and fear of reprisals is an adjustment to the social environment, involving considerations for typical human behaviour. People have a tendency to reciprocate: respond to positive and negative feedback in a like manner. This can lead to a recommender not wanting to give negative recommendations out of fear of negative consequences.

Several of the evaluated systems are capable of doing two or more different levels of reputation calculation based on the information available. If enough local experience is available, it can be used alone to calculate a reputation value. If local information alone is insufficient, recommendations are brought into the evaluation. **MDNT** and **Travos** use this approach. **REGRET** has even more levels of fallback, such as calculating the reputation of the social neighbourhood of the trustee. **Managing Trust** allows the trustor to evaluate the credibility of the mediators iteratively when the information they provide does not immediately lead into a decision. Flexibility in evaluation allows systems to balance between more accuracy and lighter calculations according to situation.

General reciprocity and fear of reprisals are only rarely considered by the systems we evaluated. The **eBay** system actively encourages users to avoid giving negative ratings before at least trying to resolve the problem offline, so the skew towards positive is known. **Managing Trust** protects against reprisals by keeping the original recommender anonymous. The authors describing **Unitec** also include suggestions for adding some privacy for the recommender. This kind of protection makes it less necessary to include considerations of the environment influencing the gathered information. **REGRET** is the only system to consider social relationships between peers when calculating the credibility of each recommendation; it uses the information for recommender selection, and to determine a neighbourhood whose members' reputation can be used to represent the reputation of a particular member for whom it is difficult to estimate directly.

7. Conclusions

Open collaborations management needs the support of robust reputation management. The survey of current reputation systems reveals various good mechanisms and methods used in the systems, but the area is still immature.

There are two points of focus for designing a reputation system to perform well in its field: *social requirements* and *scalability to real use*. We have focused on analyzing the former, as there was insufficient information available on the scalability of the analyzed systems. In addition, the simulations vary quite a bit in size and nature: **Travos** experiments with 10 nodes only, while **FuzzyTrust** trials with 10.000 peers at best. Reputation system experiments tend to focus on prediction correctness instead of performance, and no benchmarks have been established for either yet.

Social requirements depend on the application environment. The reputation-handling needs of a file sharing network and those of an electronic marketplace are different because of the varying level of inherent risk involved in a transaction. In addition, the environment of the application can be nearly closed or very open, and contain some infrastructure to support accountability or otherwise reduce risk, e.g. by insurance. Large pseudonymous networks, where the average actor is a newcomer with at most a handful of transactions, are among the most challenging environments for a reputation system. On the other hand, the application may allow us to mostly ignore all but a few actors. The challenge to a reputation system designer is to recognize what is really needed and which trade-offs are valid.

Scalability to real use requires addressing three load-related challenges: the load placed on the trustor trying to perform an analysis, the load placed on high-reputation nodes as transaction partners or recommenders, and the load placed on the network through recommendation transfers. Allowing different levels of analysis based on the situation is a partial solution to the trustor and network load. Making sure that fresh actors are regularly sought as partners and recommenders addresses the load on high-reputation nodes to a degree. Storing recommendations systematically, limiting information sources used or not distributing reputation calculations to all trustors can also address the network load problem, but demand new layers of trust.

In order to make reputation systems more mature for routine trust decisions in open network environments, a number of challenges have to be addressed. We have envisioned a global system where trust decisions are made locally, but reputation information is shared in a global reputation management network.

First, for this vision, the reputation information should be standardized to achieve interoperability between systems and services that use them. The granularity of targets to which reputation information is associated should be first determined and then, suitable identification mechanism for these targets provided. The granules of interest depend on the application area, but can involve for example humans, machines, and business services.

Second, experience-based reputation information should be based on a commonly acceptable framework of concepts, ranging for example from successful and correct performance in business transactions to illegal transactions or breaches of technical criteria. For all these axes, ontologies should be developed to capture the metrics to be used.

Third, the role we envision for reputation systems in the open collaborations creates new vulnerabilities. We have started a comprehensive threat analysis of systems supporting trust, reputation and privacy management, but additional work is still needed for creating a system that would resist these new threats. One of the essential aspects of this development is the extensive use of credibility metainformation on exchanged reputation information and development of trust decision algorithms sensitive to both the credibility measures and changes in them.

Finally, we have noted the absence of benchmarks suitable for comparing the effectiveness (performance) of making trust decisions, or causing changes in trust decisions depending on the reputation information.

References

[1] K. Aberer and Z. Despotovic. Managing trust in a peer-2-peer information system. In *Proceedings of the 10th International Conference on Information and Knowledge Management (2001 ACM CIKM)*, Atlanta, 2001.

[2] D. Chadwick. Operational models for reputation servers. In *Proceedings of Trust Management: Third International Conference (iTrust 2005)*, volume 3477 of *LNCS*, pages 108–115. Springer-Verlag, Apr. 2005.

[3] Z. Despotovic and K. Aberer. Maximum likelihood estimation of peers' performance in P2P networks. In *The Second Workshop on the Economics of Peer-to-Peer Systems*, 2004.

[4] eBay. The online marketplace, June 2006. URL `http://www.ebay.com/`.

[5] F. K. Hussain, E. Chang, and T. S. Dillon. Trustworthiness and CCCI metrics in P2P communication. *International Journal of Computer Systems Science and Engineering*, 19(3), May 2004.

[6] S. Kamvar, M. Schlosser, and H. Garcia-Molina. The EigenTrust algorithm for reputation management in P2P networks. In *Proceedings of the Twelwth International World-Wide Web Conference (WWW03)*, pages 446–458, 2003.

[7] M. Kinateder and K. Rothermel. Architecture and algorithms for a distributed reputation system. In *Proceedings of Trust Management: First International Conference (iTrust 2003)*, volume 2692 of *LNCS*, pages 1–16. Springer-Verlag, May 2003.

[8] L. Kutvonen, J. Metso, and S. Ruohomaa. From trading to eCommunity population: Responding to social and contractual challenges. In *Proceedings of the 10th IEEE International EDOC Conference (EDOC 2006)*, Hong Kong, Oct. 2006. Best paper award.

[9] S. Lee, R. Sherwood, and B. Bhattacharjee. Cooperative peer groups in NICE, 2003.

[10] M. W. Lutz Schubert et al. Trustcom reference architecture, deliverable d09. Technical report, TrustCoM WP27, Aug. 2005.

[11] J. Patel, W. L. Teacy, N. R. Jennings, and M. Luck. A probabilistic trust model for handling inaccurate reputation sources. In *Proceedings of Trust Management: Third International Conference (iTrust 2005)*, volume 3477 of *LNCS*, pages 193–209. Springer-Verlag, Apr. 2005.

[12] M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the Gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. *IEEE Internet Computing*, 6:50–57, 2002.

[13] S. Ruohomaa, L. Viljanen, and L. Kutvonen. Guarding enterprise collaborations with trust decisions—the TuBE approach. In *Proceedings of the First International Workshop on Interoperability Solutions to Trust, Security, Policies and QoS for Enhanced Enterprise Systems (IS-TSPQ 2006)*, Mar. 2006.

[14] J. Sabater and C. Sierra. Reputation and social network analysis in multi-agent systems. In *AAMAS '02: Proceedings of the First International Joint Conference on Autonomous Agents and MultiAgent Systems*, pages 475–482, 2002.

[15] S. Song, K. Hwang, R. Zhou, and Y.-K. Kwok. Trusted P2P transactions with fuzzy reputation aggregation. *IEEE Internet Computing*, 9(6):24–34, 2005.

[16] S. Staab et al. The pudding of trust. *IEEE Intelligent Systems*, 19(5):74–88, 2004.

[17] L. Xiong and L. Liu. PeerTrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transactions on Knowledge and Data Engineering*, 16(7):843–857, 2004.