Trust Transferability Among Similar Contexts[*]

Mozhgan Tavakolifard
Center for Quantifiable Quality of Service in Communication Systems
Norwegian University of Science and Technology
Trondheim, Norway
mozhgan@q2s.ntnu.no

Svein J. Knapskog
Center for Quantifiable Quality of Service in Communication Systems
Norwegian University of Science and Technology
Trondheim, Norway
knapskog@q2s.ntnu.no

Peter Herrmann
Telematics Department
Norwegian University of Science and Technology
Trondheim, Norway
herrmann@item.ntnu.no

ABSTRACT

Trust is a fundamental concern in electronic transactions and behavior of people are influenced by the situation. Motivated by that we present a state-of-the-art survey of context representation in trust management and provide main directions along which research efforts have been done. We propose a generalized model which considers different aspects of the relationship between context-awareness and trust management.

Categories and Subject Descriptors

K.6.5 [**Management of Computing and Information Systems**]: Security and Protection

General Terms

Security

Keywords

Trust, Context, Ontology, Semantic Similarity

1. INTRODUCTION AND MOTIVATION

While the need to consider a plurality of aspects as a basis for trust decisions has been recognized for a long time, the context issue has been long neglected by the trust research community [15]. The majority of trust models have considered two factors in order to estimate the value of trust in the next interaction: history of relationship and recommendations. The former is based on direct previous experiences between truster and trustee while the latter is information which truster receives about trustee from others. Furthermore, the following factors can be considered in order to improve our estimations:

- Certificates: A trusted third party might vouch for the trustworthiness, for example a banking employee is trustworthy with your bank account, or a police officer can stop your car [10].
- Self representation: This factor involves all the information a trustee gives about herself, for instance her resume or the appearance of a physical building for a company.
- Common grounds : We tend to trust people that have a common ground with us, for example family, work colleagues, church community, people from the same village, hobby club etc [10].

These three factors show how context information can enhance trust establishment. The interaction between trust and context has attracted the attention of researches only recently, and from various perspectives. By analyzing the state of the art in trust models we can see that most models do not take into account the fact that interactions take place within a particular organizational and environmental *context*. Context representation in trust models is a key issue that need to be solved in order to have a comprehensive trust model. By the word *context* we mean the same as the widely accepted definition of context [6]:

"Context is any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves."

Our motivation to consider context in trust evaluation is its ability to bring additional knowledge to the reasoning process and thus focus attention on relevant details. This paper proposes an extended state of the art analysis and a model to analyze correlation of trust information between different contexts and a mechanism to initialize reputation of nodes in unanticipated contexts (e.g. if a person is trusted in academia he will most likely trusted in an industrial arena as well).

The rest of this paper is organized as follows. A comprehensive survey of relevant related work is presented in Section 2 in order to pinpoint the contribution of this paper. In section 3 our proposed model is presented. Some application examples are given to demonstrate the potential uses of our proposed model in Section 4. Finally, Section

[*] *"Center for Quantifiable Quality of Service in Communication Systems, Center of Excellence" appointed by The Research Council of Norway, funded by the Research Council, NTNU and UNINETT. at* `http://www.q2s.ntnu.no`

Q2SWinet'08, October 27–28, 2008, Vancouver, BC, Canada.
Copyright 2008 ACM 978-1-60558-237-5/08/10 ...$5.00.

5 concludes the paper and outlines some future issues concerning the applicability of the proposed method.

2. RELATED WORK

From the literature we can find that extension of a trust model with context representation can

- Reduce complexity in management of trust relationships [13]
- Improve the recommendation process [14]
- Help to infer trust information in context hierarchies [10]
- Improve performance [15]
- Help to learn policies/norms at runtime [15, 20]
- Provide protection against changes of identity and first time offenders [15, 16]

and context related information has been represented in following ways

- Context-aware domains [13] and [14]
- Intensional Programming [23]
- Multi-dimensional goals [9]
- Clustering [15]
- Ontologies [20]

In the rest of this section different approaches to this problem are examined in more detail.

Neisse et al. [13] proposed the idea of using the abstraction of context-aware domains to reduce the complexity in the management of trust relationships. In a large context-aware system, with thousand of components and users, trust relationships can not be associated with individual entities, as this can easily become unmanageable. Examples of context-aware management domain definitions are "Nearby persons", "Personal devices",and "Working colleagues". This is the same as the common ground concept introduced earlier. The idea is to provide mechanisms to define and infer the trust degree of an entity based on the context information provided about that entity. According to Neisse et al. [14] it is also possible to use context information to improve the recommendation process (to determine from whom to request recommendation). This will allow anonymous and still useful recommendations exchange.

In [10] it is noted that context can often be structured hierarchically. For example, if you trust someone to drive your car, then you would most likely also give him your car keys or the keys to the garage .Therefore, it is necessary to identify possible hierarchical structures between different contexts in our model to be able to infer trust information from one into the other. In this work, entities, which can be applications, other users or agents that act on behalf of users are structured into a context-based trust graph. The position in this graph indicates the context-based trust level and changes based on events or over time. The structure of the trust graph reflects a certain hierarchy.

Alagar et al. [1] investigated the intensional programming paradigm for agent communication by introducing context as a first class object in the intensional programming language Lucid. Intensional programming is a powerful and expressive paradigm based on Intensional Logic. Intensional logic is a branch of mathematical logic used to precisely describe context-dependent entities. In this paper definitions, syntax, and operators for context, and introduces an operational semantics for evaluating expressions in extended Lucid are given. It is demonstrated that the extended Lucid language, called Agent Intensional Programming Language(AIPL), has the generality and the expressiveness for being an Agent Communication Language(ACL). Based on this work a context-specific trust model for multi-agent systems is introduced in [23]. The explicit introduction of context in the computation of trust, annotating trust policies with context conditions, and defining delegation through related contexts are some of the new results given in this paper.

The context issue has been viewed as multi-dimensional trust modeling for agents when goal requirements are multi-dimensional in [9]. An agent's reward is determined both by goal requirements and behavioral constraints of potential partners (e.g. quality, timeliness, availability, and cost).

In [19] authors propose an algorithm to estimate trust when truster and trustee are completely unfamiliar with each other. According to their algorithm the truster uses her past experiences which occurred in the same context as the current context to form a training set. Then using maximum likelihood estimation method, the trust value for a new trustee can be estimated.

Rehak et al. [15] define a set of reference contexts in a metric space and associate truthfulness data with it. These data are updated and queried with weight that decreases with distance between the current situation and the reference context. The model uses *Leader-Follower* clustering to identify the reference contexts to be representative of the data (The advantages of this clustering method is that it allows an on-line approach, without pre-specifying the number of expected clusters and requires only a single parameter as its input. The biggest disadvantage is that it may easily under or over estimate the number of clusters.). In an empirical test, it is shown that context-aware models easily outperform general trust models when the situation has an impact on partner trustfulness and that their performance and efficiency is comparable with general trust models where the trustfulness is independent of the situation. In this work two advanced uses of representing context for multiagent trust modeling is proposed: (i) policy/norm learning at runtime by analyzing data regarding the performance of different agents in similar situations (e.g. when all agents fail in a certain situation, they may agree to introduce a policy that specifically prohibits such actions) (ii) reasoning based on uncertain identities by decomposing the single identity dimension into an identity subspace, where each agent is defined by one or more crucial properties. With this modification, the trust model can make predictions about the performance of agents by exploiting data characterizing similar agent's performance in the past. The main advantages are that the extended model learns faster and once the new agent is categorized, its performance can be predicted. This is also a clear advantage in ad-hoc environments, where there is no agent platform to enforce unique identity.

Based on this model, Rehak et al. [16] conclude that the extension of a trust model with a context representation environment can be extended to encompass a more open situation (e.g. a wireless sensor network that is hard to identify and where the barriers of entry are quite low). In such environments it is not needed to have assumptions like:(i)proven identity, (ii)repetitive interactions and (iii)similar trusting situations. The fact that two agents with presumably distinct identities can be considered identical by a context-sensitive trust model may provide protection against changes of identities. This approach is also effective against first time offenders (we can obtain a model with inductive properties, able to estimate the performance of new entrants using the experience with the similar partners in the past).

Golbeck et al. [8] have proposed an ontology for trust. In [7] they have considered a model using context-specific reputation by assigning numeric ratings to different types of connections based on the context of the analysis. In [20] rules to describe how certain context-sensitive information (trust factors) reduces or enhances the trust value have been specified for this trust ontology. The authors also argue that a specific advantage of making the context explicit in message exchanges is that this information can be used in trust policies. For example, a policy can state that news information related to a particular location is to be trusted more if the reporting entity was at the location at the time when the event occurred. In this sense, policies define how to process context information to derive trustworthiness assertions. But they have not answered how the context-sensitive trust factor should be determined. In addition they have not addressed the fact that the trust factor might be different for different aspects of trust.

In [11] trust is formalized by using situation calculus in order to define a trust ontology. Situation calculus is a logic language specifically designed for representing dynamically changing worlds. It works in the following way: the changing world is represented by a set of fluents. A fluent is a property (of the world) whose value is dependent on situations. In other words, a fluent dynamically changes when the situation does. The situation, in turn, changes when an action is performed by agent(s) in the world. Trust and context are represented as fluents.

In [21] contextual information (context attributes) is used to adjust the output of a trust determination process. Each attribute can adjust the trust value positively or negatively according to a specified weight. For example if t is the trust value and ω is the weight of the context property then the adjusting function can be t^{ω} for decrease or $\sqrt[\omega]{t}$ for increase. A context ontology connects the context attributes with each other in an appropriate manner, enabling the utilization of context attributes which do not exactly match the query, but are “close enough” to it. For example, the QoS properties of a network, over which some software component is downloaded, can be described in such an ontology (Fig.1). Suppose that the current network (B_1) is not pre-evaluated with regard to its impact on trustworthiness. However, as its neighbors in the ontology are networks which have pre-evaluated trustworthiness values (B_2, U, and G). By using these values as well as their “semantic distance” to the current network, the resulting trustworthiness can be estimated. The semantic distance is calculated by taking into account the “upwards cotopy”, that is, the distance between the currently investigated concept and a root-concept of the ontology. The upwards cotopy is calculated as the ratio between the number of shared nodes from the source node and the sink node to the root node, and the total number of nodes from the source and the sink to the root node. For example, in the case of B_1 and B_2, the numbers are $|Bluetooth, PacketSwitched, Wireless, Network| = 4$ and $|B1, B2, Bluetooth, PacketSwitched, Wireless, Network| = 6$ and the semantic distance between the source and the sink therefore is $\frac{4}{6} \approx 0.67$. If adjustment functions for B2, U, and G are $\sqrt[\omega_1]{t}$, $\sqrt[\omega_2]{t}$, and t^{ω_3} and their semantic distances to B_1 are d_1, d_2, and d_3 respectively then our estimate of adjusting function for B_1 will be $\sqrt[\omega_1 * d_1]{\sqrt[\omega_2 * d_2]{t^{(\omega_3 * d_3)}}}$.

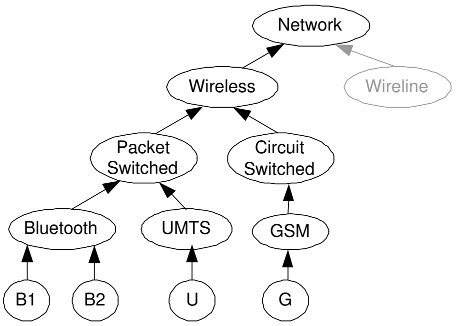

Figure 1: Concepts in the network ontology [21].

In this work the notion of context also has been applied to the reputations by emphasizing more the observations that have taken place under similar conditions as where the truster currently is. Two relationships have been considered between recommendations and context. First, as was the case with reputation, the contextual details at the time when the recommendation was made can be considered and compared with the truster's current context. Note that considering this is not as straightforward as was the case with reputation, since recommendations come from others, not from the truster. Secondly, also the recommendation content can be context-dependent.

In [5] cases where an agent does not have enough information to produce a trust value for a given task, but she knows instead the previous partner behavior performing similar tasks are considered. This model estimates trust using the information about similar tasks. The similarity ($D(s_1, s_2)$) between two tasks s_1 and s_2 is obtained from the comparison of the task attributes.

$$D(s_1, s_2) = 1 - \frac{1}{n}.\sum_{i=1}^{n} |s_{1_i} - s_{2_i}|$$

where n is the number of task attributes, s_{1_i} is the $i - th$ attribute of task s_1, and s_{2_i} is the $i - th$ attribute of task s_2.

The same authors in [4] obtain the similarity ($D(s_1, s_2)$) from the comparison of the task attributes in the ontology using formula below:

$$\frac{|S_1 \cap S_2|}{|S_1 \cap S_2| + \alpha(s_1, s_2)\,|S_1 \backslash S_2| + (1 - \alpha(s_1, s_2))\,|S_2 \backslash S_1|}$$

where $0 \prec \alpha \prec 1$; S_1 and S_2 are the set of properties of concepts s_1 and s_2, respectively. Function α takes into account the depth of compared concepts in the ontology hierarchy. In [22] a model with name of CAT (A Context-Aware Trust Model) using some keywords to describe contexts is pro-

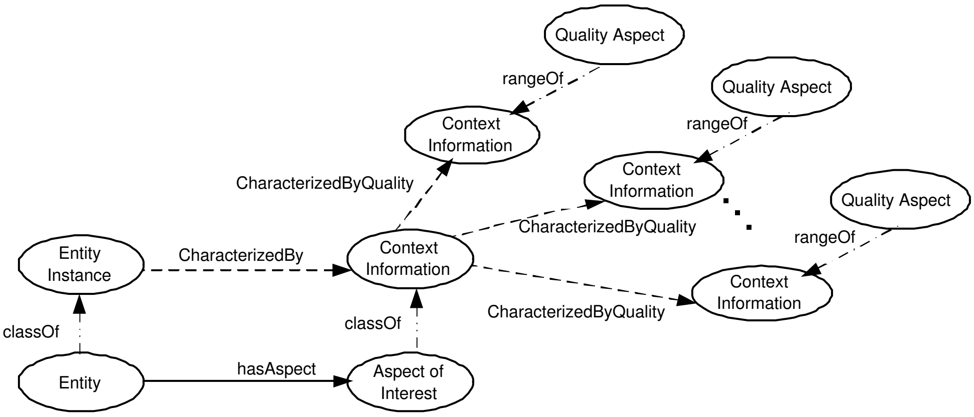

Figure 2: ASC Model [18].

posed. The similarity between two contexts with K_1 and K_2 as sets of keywords is calculated by $\frac{K_1 \cap K_2}{K_1 \cup K_2}$.

Bagheri et al. [3] have proposed a framework for dynamically updating and inferring the unobserved reputation of environment participants in different contexts. This framework proposes the employment of a reputation structure tree to represent the relationship between the contexts of the environment. Reputation of a given identity in one context can be propagated to other contexts through two mechanisms, namely: forward update and backward adjustment. This work does not mention how to develop the reputation structure tree. Bagheri et al. [2] also propose a framework for their previous proposal based on valuation networks. Global reputation is modeled as Dempster-Shafer belief functions on a Markov tree through which the relationship between various contexts of a unique environment are modeled through hyper-vertices of the Markov tree. Reputation of each identity in a given context is represented using a belief mass assignment function. The estimation of reputation in various contexts of the environment is performed by the employment of the message passing-based belief propagation model of the Shenoy-Shafer architecture.

3. THE ENHANCED TRUST MODEL

As many researchers realized, trust may be context specific, for example, a person may trust her or his financial advisor about investment analysis but does not trust the same advisor in health-care. The context in which the truster is confronted with the trust judgment problem might not be the same as the context in which the expected valid information has been created. For example, an investor (the truster) attempts to use some information in context of buying stocks that has been created by a financial expert (the trustee) in another context of giving a financial investment seminar [11]. Therefore we need a suitable means to represent context for trust evaluation and a suitable method to infer trust information between different contexts.

We consider our approach in this paper as a complementary solution in comparison with [21] and [4]. In [21] nothing is said about how many nodes are included or what other context dependent parameters should be included in the calculation. This work does not mention how to find similar or relevant nodes. The main drawback of [4] is that does not say anything about how to find similar or relevant contexts.

[17] provides a survey of different approaches to model context for ubiquitous computing. In this work numerous approaches are reviewed, classified relative to their core elements and evaluated with respect to their appropriateness for ubiquitous computing. The authors at the conclusion that the most promising assets for context modeling for ubiquitous computing environments can be found in the ontology category in comparison with other approaches like key-value models, mark-up scheme models, graphical models, object-oriented models, and logic based models. This selection is based on the six requirements dominant in pervasive environments: distributed composition, partial validation, richness and quality of information, incompleteness and ambiguity, level of formality, and applicability to existing environments.

We propose an alternate model based on the *Aspect-Scale-Context (ASC) model* which is an ontology-based context model introduced in [18] (Fig.2). This model may be used to describe contextual facts and contextual interrelationships in a precise and traceable manner and thus may be engaged to determine contextual interoperability. It has an ontology reasoner to determine interrelationship dependencies and relevance conditions. The main contribution of our work is that we improve the ASC model by the inclusion of a new similarity measurement algorithm and use the relation between contexts not only for bootstrapping when the context is unfamiliar but also for propagation of information among contexts.

We define our terminology according to [18]:

- A *context information* is any information which can be used to characterize the state of an entity concerning a specific aspect.
- An *entity* is a person, a place or an object.
- An *aspect* is a classification, symbol- or value-range, whose subsets are supersets of all reachable states.
- A *context* is the set of all context information characterizing the entities relevant for a specific task with their relevant aspects.
- An *entity is relevant* for a specific task if its state is characterized at least concerning one relevant aspect.

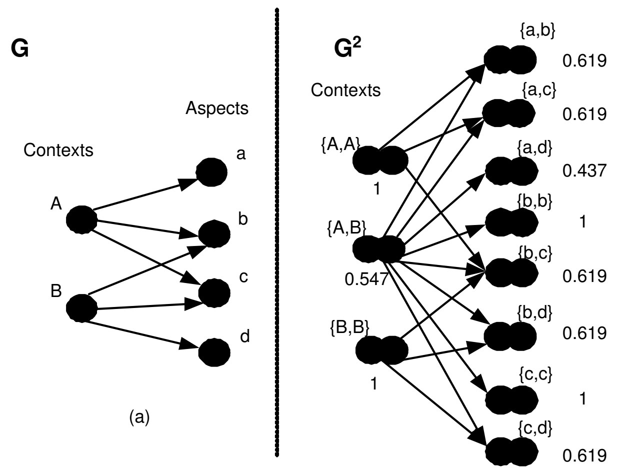

Figure 3: Graph model of context information in the ASC ontology model.

- An *aspect is relevant*, if the state with respect to this aspect is accessed during a specific task or the state has any kind of influence on the task.
- A *situation* is the set of all known context information.

In order to measure similarity among contexts we use the idea of the bipartite SimRank which is an extension of the basic SimRank algorithm [12] to bipartite domains consisting of two types of objects. SimRank is a general similarity measurement algorithm of objects applicable in any domain with object-to-object relationships. Such domains are naturally modeled as graphs, with nodes representing objects and edges representing relationships. Therefore, we need to form a graph with contexts and aspects as nodes. In this graph each context points to their aspects Fig.3(a). The recursive intuition behind this algorithm is that in many domains, *similar* objects are related to *similar* objects. More precisely, contexts A and B are similar if they are related to aspects b and c, respectively, and b and c are themselves similar. The base case is that aspects are similar to themselves.

A general comparator can be defined for an aspect (for each scale) to measure similarity among values of the same aspects (in the base case). These comparators has one of the following properties:

- *Categorical*: values in the same category are more similar (e.g., weather).
- *Continuous*: closer values are alike (e.g., temperature).
- *Hierarchical*: a more general context can be used when no trust information for a specific context are available (e.g., location).

Aspects which do not have these characteristics may require a custom comparator to be defined for them.

If we call the graph of contexts and their relations G, then we can form a node-pair graph G^2 in which each node represents an ordered pair of nodes of G Fig.3(b). A node (A, B) of G^2 points to a node (a, b) if, in G, A points to a and B points to b. Similarity scores are symmetric, so for clarity we draw (A, B) and (B, A) as a single node A, B (with the union of their associated edges). For a node v in G^2, we denote by $I(v)$ and $O(v)$ the set of in-neighbors and out-neighbors of v, respectively. Individual neighbors are denoted as $I_i(v)$, for $1 \leq i \leq |I(v)|$, and individual out-neighbors are denoted as $O_i(v)$, for $1 \leq i \leq |O(v)|$.

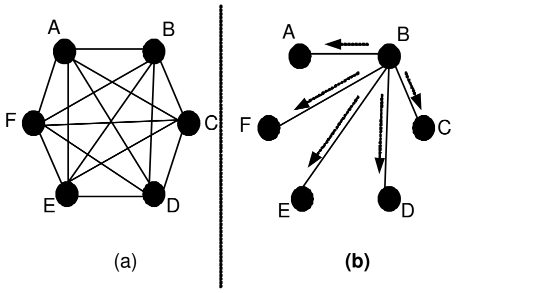

Figure 4: Update propagation among contexts.

SimRank is an iterative fixed-point algorithm on G^2 to compute similarity scores for node-pairs in it. The similarity score for a node v of G^2 gives a measure of similarity between the two nodes of G represented by v. Scores can be thought of as flowing from a node to its neighbors. Each iteration propagates scores one step forward along the direction of the edges, until the system stabilizes (i.e., scores converge). Since nodes of G^2 represents pairs in G, similarity is propagated from pair to pair. Under this computation,

- Two contexts are *similar* if they have (in the graph points to) *similar* aspects.
- Two aspects are *similar* if they belong to (are pointed by) *similar* contexts.

Fig.3(a) shows a sample graph with two contexts, four aspects and the relationships among them. Fig.3(b) shows the derived node-pair graph G^2 for the graph G in Fig.3(a). Similarity scores for nodes of G^2, computed using $C_1 = C_2 = 0.8$, are also shown.

Let $s(A, B)$ denote the similarity between contexts A and B, and let $s(c, d)$ denote the similarity between aspects c and d.Since directed edges go from contexts to aspects, for contexts $A \neq B$ we have the following recursive equation:

$$s(A, B) = \frac{C_1}{|O(A)|\,|O(B)|} \sum_{i=1}^{|O(A)|} \sum_{j=1}^{|O(B)|} s(O_i(A), O_j(B)) \quad (1)$$

where C_1 is a constant between 0 and 1. If $A = B$ then $s(A, B)$ is defined to be 1. (1) says that the similarity between contexts A and B is the average similarity between the aspects they have. $s(A, B)$ is 0 when $I(A) = 0$ or $I(B) = 0$.

In case of aspects $c = d$ we use the comparator function explained earlier for $s(c, d)$. Otherwise,

$$s(c, d) = \frac{C_2}{|I(c)|\,|I(d)|} \sum_{i=1}^{|I(c)|} \sum_{j=1}^{|I(d)|} s(I_i(c), I_j(d)) \qquad (2)$$

C_2 is a constant between 0 and 1. (2) says that the similarity between aspects c and d is the average similarity between the contexts they belong to. C_1 and C_2 can be thought of either as confidence levels or decay factors. Consider a

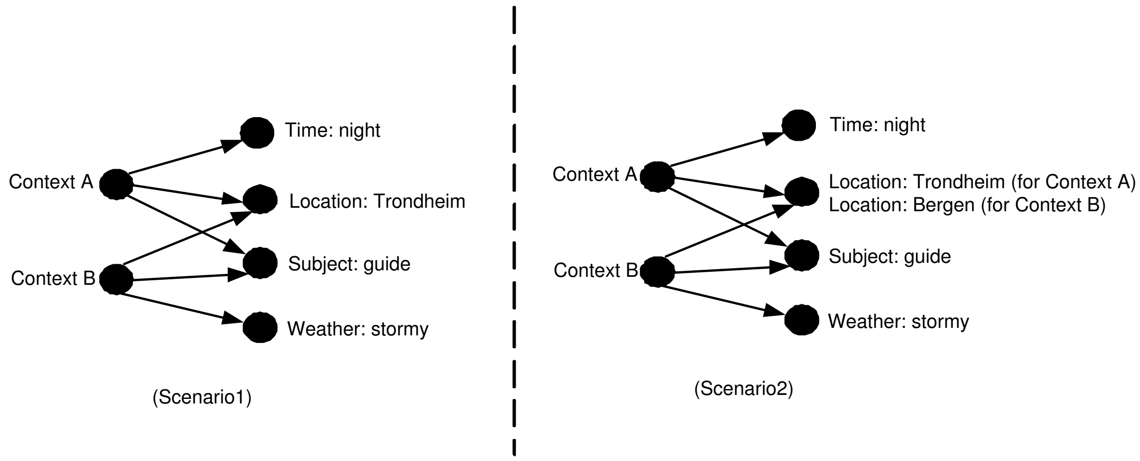

Figure 5: Application example.

simple scenario where context X has two aspects c and d, so we conclude that there is some similarity between c and d. The similarity of x with itself is 1, but we probably do not want to conclude that $s(c,d) = s(x,x) = 1$. Rather, we let $s(c,d) = C_2.s(x,x)$ meaning that we are less confident about the similarity between c and d than we are between x and itself.

A complete graph with contexts as nodes and measure of similarity of every two nodes as weight of the edge between them can be formed. A spanning tree of this graph with the node that has been updated as its root can be employed to propagate an update in one context to other contexts. Fig.4(a) shows the complete graph for six contexts and Fig.4(b) shows the resulting spanning tree from a change in value of context B. This change multiplied by the similarity measure on each edge will be added to the value (the trust value associated for each agent in each context) of other contexts. Arrows show the direction of propagation.

4. APPLICATION EXAMPLE

In the following, to demonstrate the potential uses of the proposed context aware model in trust judgments, we apply this model to trust judgment problem in some motivating scenarios.

Scenario 1: Alice is wondering about trusting Bob to guide her in Trondheim at night. Consider this case as *context A* and *Location: Trondheim*, *Time: night*, and *Subject: guide* as corresponding aspects [1]. Assume that the trust model (one of available trust estimation models) returns the trust value of "very trustworthy" among the five possible trust values of "very untrustworthy", "untrustworthy", "moderate", "trustworthy", and "very trustworthy" (equivalent to +2 out of -2, -1, 0, +1, +2) based on available information. Consider another case (context B) that Alice is going to trust Bob to guide her in Trondheim when it is stormy (aspects are: *Location: Trondheim*, *Subject: guide*, and *Weather: stormy*). Based on the proposed model we can infer an initial trust value of "trustworthy" ($+2 \times 0.547 = 1.094 \approx +1$) for this case. The calculation details is shown in Fig.5.

Scenario 2: this scenario is similar to previous one, but in latter case Alice is going to trust Bob to guide her in Bergen when it is stormy. Context A and Context B have the aspect of *Location* in common but the values are different. In this case we should use the comparator function of the *Location* aspect in order to compute the similarity between the two values. Therefore the resulting similarity value between two contexts will be less.

5. CONCLUSION AND FUTURE WORK

In this paper we propose a model that clearly depicts how trust information in one context can affect trust information in other contexts. This model also provides suitable mechanisms to anticipate a proper initial reputation value for a trustee within contexts that she has not been present before. The main contributions of this paper are as follows:

- A comprehensive survey of context-sensitive trust management.
- An improvement of the ASC model by inclusion of a similarity measurement algorithm.
- Inclusion of comparator function in the SimRank algorithm.
- Representation of inter-relations among different contexts based on the semantic similarity of current trust models.

In the future, we plan to implement our model, do simulations and probably use data from a real system to examine how well it works. The SimRank similarity measurement algorithm fits the ASC model because of its structure. But still we need to compare it with other similarity measurement algorithms that have been proposed for ontologies. Furthermore, in more complicated cases, the similarity of two context models is in itself context dependent. Consequently, we have to take this into account when evaluating the similarity between contexts.

[1] *The Context Management Access Point (CMAP)* interface [18] specifies the relevant aspects of interest using the Context Binding module

6. REFERENCES

[1] V. Alagar, J. Paquet, and K. Wan. Intensional Programming for Agent Communication. In *Proceedings of DALT*, volume 4, pages 239–255. Springer.

[2] E. Bagheri, M. Barouni-Ebrahimi, R. Zafarani, and A. Ghorbani. A Belief-Theoretic Reputation Estimation Model for Multi-Context Communities.

[3] E. Bagheri and A. Ghorbani. Behavior analysis through reputation propagation in a multi-context environment. *International Conference on Privacy, Security and Trust (PSTŠ06)*, 2006.

[4] A. Caballero, J. A. B. Blaya, and A. F. Gómez-Skarmeta. A new model for trust and reputation management with an ontology based approach for similarity between tasks. In *MATES*, pages 172–183, 2006.

[5] A. Caballero, J. Botia, and A. Gomez-Skarmeta. On the Behaviour of the TRSIM Model for Trust and Reputation. *LECTURE NOTES IN COMPUTER SCIENCE*, 4687:182, 2007.

[6] A. Dey. Understanding and Using Context. *Personal and Ubiquitous Computing*, 5(1):4–7, 2001.

[7] J. Golbeck and J. Hendler. Inferring Reputation on the Semantic Web. In *Proceedings of the 13th International World Wide Web Conference*, 2004.

[8] J. Golbeck, B. Parsia, and J. Hendler. Trust Networks on the Semantic Web. In *Proceedings of Cooperative Intelligent Agents*, volume 2003. Springer, 2003.

[9] N. Gujral, D. DeAngelis, K. Fullam, and K. Barber. Modeling Multi-Dimensional Trust. In *the Proceedings of the Workshop on Trust in Agent Societies*, pages 8–12.

[10] S. Holtmanns and Z. Yan. Context-Aware Adaptive Trust.

[11] J. Huang and M. S. Fox. An ontology of trust: formal semantics and transitivity. In *ICEC '06: Proceedings of the 8th International Conference on Electronic Commerce*, pages 259–270, New York, NY, USA, 2006. ACM.

[12] G. Jeh and J. Widom. SimRank: a measure of structural-context similarity. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 538–543. ACM Press New York, NY, USA, 2002.

[13] R. Neisse, M. Wegdam, and M. van Sinderen. Context-Aware Trust Domains. *1st European Conference on Smart Sensing and Context, Enschede, The Netherlands, Oct-2006*, 2006.

[14] R. Neisse, M. Wegdam, M. van Sinderen, and G. Lenzini. Trust Management Model and Architecture for Context-Aware Service Platforms. *LECTURE NOTES IN COMPUTER SCIENCE*, 4804:1803, 2007.

[15] M. Rehak, M. Gregor, M. Pechoucek, and J. Bradshaw. Representing Context for Multiagent Trust Modeling. In *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT 2006 Main Conference Proceedings)(IAT'06)-Volume 00*, pages 737–746. IEEE Computer Society Washington, DC, USA, 2006.

[16] M. Rehak and M. Pechoucek. Trust modeling with context representation and generalized identities. *Klusch, M., Hindriks, K., apazoglou, MP, Sterling, L.(eds.) CIA*, pages 298–312, 2007.

[17] T. Strang and C. Linnhoff-Popien. A context modeling survey. *Workshop on Advanced Context Modelling, Reasoning and Management as part of UbiComp*, 2004.

[18] T. Strang, C. Linnhoff-Popien, and K. Frank. CoOL: A Context Ontology Language to enable Contextual Interoperability. *LNCS*, 2893:236–247, 2003.

[19] M. Tavakolifard and S. Knapskog. Probabilistic Reputation Algorithm for Decentralized Multi-Agent Environmentsc. In *Proceedings of the fourth international workshop on security and trust management (STM 08)*, 2008.

[20] S. Toivonen and G. Denker. The impact of context on the trustworthiness of communication: An ontological approach. In *Proceedings of the Trust, Security, and Reputation on the Semantic Web Workshop, held in conjunction with the 3rd International Semantic Web Conference (ISWC 2004), Hiroshima, Japan*, volume 127, 2004.

[21] S. Toivonen, G. Lenzini, and I. Uusitalo. Context-aware trust evaluation functions for dynamic reconfigurable systems. In *Proceedings of the Models of Trust for the Web Workshop (MTWŠ06), held in conjunction with the 15th International World Wide Web Conference (WWW2006) May*, volume 22, 2006.

[22] M. Uddin, M. Zulkernine, and S. Ahamed. CAT: a context-aware trust model for open and dynamic systems. In *Proceedings of the 2008 ACM Symposium on Applied Computing*, pages 2024–2029. ACM New York, NY, USA, 2008.

[23] K. Wan and V. Alagar. An Intensional Functional Model of Trust. *Der Leistungsbedarf und seine Deckung: Analysen U. Strategien: VDI-VDE-gfpe-tagung in Schliersee am 16.-17. Mai 1979*, 1979.