Transaction Similarity-Based Contextual Trust Evaluation in E-Commerce and E-Service Environments

Haibin Zhang, Yan Wang
Department of Computing
Macquarie University
NSW 2109, Australia
{*haibin.zhang, yan.wang*}*@mq.edu.au*

Xiuzhen Zhang
School of Computer Science and IT
RMIT University
Melbourne, Victoria 3001, Australia
xiuzhen.zhang@rmit.edu.au

***Abstract*—The trust of sellers and transactions is a very important issue in e-commerce and e-service environments. At some e-commerce websites (such as eBay[1]), the trust management mechanism can compute a trust value of a seller, which is based on the ratings of past transactions given by buyers. This trust value, however, is static and can only reflect the general or global trust status of a seller, and it is not directly bound to a new transaction. As a result, a buyer may be easily cheated by a malicious seller in a new transaction with the notorious value imbalance problem [5], i.e., the malicious seller can build up a good reputation by selling cheap products/services and then start to cheat buyers by selling expensive products/services. Instead of providing such a static trust value, in order to provide more objective trust result for a new potential transaction, a trust evaluation mechanism should based on the ratings of past transactions, the nature of both past transactions and the new transaction. In this paper, we propose a new contextual trust evaluation method. Our method compares the transaction context similarity between the new transaction and past transactions, from which the trust value of the new transaction can be determined. Our method can identify and prevent potentially malicious transactions with the value imbalance problem.**

***Keywords*-Transaction similarity; Contextual trust; E-Commerce; E-Service;**

I. Introduction

Some e-commerce websites (such as eBay) have introduced simple trust management mechanisms to provide valuable trust information to buyers before making payment. After each transaction, a buyer has the opportunity to give a rating (the rating can be "positive", "neutral" or "negative") to the e-commerce system according to the service quality of the seller, and then these ratings over a recent period are accumulated and a single positive feedback rate (i.e. 98% or 99%) is computed from them to reflect the trust status of the seller in past transactions.

Such a trust value is used to represent the reputation of a seller, but it only reflects the *general* trust status of this seller. While a buyer is more concerned about the trust status of a new transaction for the products/services that s/he is going to order, the buyer can hardly rely on this static trust value, which does not directly infer the trust level of the forthcoming transaction, and it is static to the new transactions of selling different products/services. As a result, with such a simple trust management mechanism, buyers are vulnerable to some frauds from malicious sellers. For example, a malicious seller can build up a good reputation by selling cheap products. After having obtained a good reputation, the seller can start cheating buyers by selling expensive products [21], [22]. Kerr *et al*. named this attack as a *value imbalance problem* [5], and several real world cases with this problem have been reported [12]. For instance, an Australian reporter reported having tracked down a fraudster at eBay who tricked over 130 people for more than AU$\$10,000$. The fraudster traded genuinely by selling cheap products to build up a positive profile before committing fraud by selling expensive products. A Californian fraudster who used the name "*kuchar1*", in the same way, managed to earn a high positive feedback rate and defrauded buyers for over US$\$300,000$.

Actually, this problem is due to the independence between trust evaluation and contextual information in transactions. Generally speaking, most trust evaluation models mainly consider two factors: direct experience between a trustor (the subject that trusts a target entity) and a trustee (the entity that is trusted), and recommendations from others. Trust evaluation from either direct experience or recommendations from others, however, is different in different contexts. When a seller begins to sell different products/services, its previous trust value provided by a trust management mechanism can not represent the seller's trust status in a forthcoming transaction, as trust is context dependent [7], [8].

The interaction between trust and context information has already attracted the attention of researchers from various disciplines. Some social scientists like McKnight *et al*. [8] have proposed *interpersonal and personal trust* as one of topological categories on trust, namely, one person trusts another person in a specific situation. For example, Alice may trust Bob as a mechanic in the specific context of servicing her car but probably not in the context of babysitting her children [1]. Similarly, from the computer science

[1] www.ebay.com

discipline, Marsh is the pioneer to propose the concept of *situational trust*, which is described in an example. "Whilst I may trust my brother to drive me to the airport, I most certainly would not trust him to fly the plane!" [7] That is to say, even the the same person, different situations will require different considerations with regard to trust. Rehak *et al*. [10] pointed out the difference between *context* and *situation* is that situation is the state of reality and context is a formal, simplified representation of the situation.

In contrast with a static trust value to present the *general* trust status of a seller at some e-commerce websites, a good trust management system should provide trust information that indicates not only the trust level of past transactions, but also the trust status associated with each forthcoming transaction.

In this paper, we propose a contextual transaction trust evaluation method in e-commerce and e-service environments, which is associated with both past transactions and a forthcoming transaction. In order to infer the trust level of a forthcoming transaction, we compare the context similarity between past transactions and the new transaction. Our work do not explicitly differentiate e-commerce and e-services environments as they both have the same problems and the same needs in contextual transaction trust computation. Furthermore, considering that there are many contextual properties in transactions, our approach mainly focuses on product/service category and transaction amount (i.e. product price). We also introduce an example to illustrate how our proposed method works.

The paper is organized as follows. Section II introduces related work. Section III presents our transaction similarity-based contextual trust evaluation method. An application example is given in Section IV to demonstrate our proposed method. Finally, section V concludes this paper.

II. Related Work

In the literature, there are some studies considering the relationship between trust evaluation and context information.

A. Trust Evaluation Based on Multi-dimensional Transaction Attributes

Griffiths [3] proposes a Multi-Dimensional Trust (MDT) model, based on *general trust* described by Marsh [7], but multi-dimensional trust is distinct from both *situational trust* and *general trust*. The author takes transaction attributes (e.g., timeless, quality and cost) into account and let a buyer specify the weights of attributes for trust computation. Thus given the same seller, the trust results computed for different buyers may be different. In the MDT model, buyers use their own direct experience to evaluate sellers. Similarly, in the REGRET system, Sabater *et al*. propose that reputation of sellers can also be measured from different dimensions. Buyers also weigh these different dimensions using his own experience and computer a "general reputation" [13]. However, as Griffiths himself pointed out, MDT could only be regarded as the complementary to "general trust". It is because that buyers could only get trust value of a seller in a particular situation from different dimensions [3]. That is to say, the buyers know the trust status of a seller selling *design clothes* from several dimensions (i.e. the quality of clothes, timeless to handle transactions), but they are still not sure about whether they could trust this seller when s/he begins to sell *printers*. This model still cannot prevent the *value imbalance problem*.

B. Trust Evaluation Based on Context Inheritance (CI)

Some researchers, from another angle, propose a conception of trust inherited to deal with the problem of contextual trust evaluation. Holtmanns *et al*. conceptually pointed out that context can often be structured hierarchically. For example, if I trust my brother to drive me to the airport, I can most likely give him my car keys, as giving him my car keys can be considered as a subset of the access rights of driving my car [4]. Samek *et al*., likewise, propose a hierarchical model of trust in contexts (HMTC), which is used to find inclusion relations between contexts [14]; hence identifying possible hierarchical structures between different contexts can assist us to infer the trust value from one to another. Though their task is different from ours, hierarchy in a contextual property (e.g., product category is a hierarchy as a tree structure) has been considered.

C. Trust Evaluation Based on Context Similarity (CS)

Mui has the following description on context-based trust evaluation. Consider part A in the rating model (like eBay) who has never interacted with part B in the past. Before taking the trouble of interacting with B, A asks other parties in the same context what their ratings for B are. Trust will be established between A and B if the weighted sum of the ratings from other entities is greater than a certain threshold. The weights on the ratings from other agents are determined by how uniform the environment is with A [9]. Following the above analysis, to evaluate the trust level of a forthcoming transaction, a buyer rely on the ratings from other buyers, who traded with the same seller before, and then compare *context similarity* (CS) between two buyers. Therefore, *context similarity* (CS) calculation is regarded as an important means to deal with contextual trust evaluation problem. With respect to the context description methods, existing studies can be divided into two different categories.

1) Context Description Based on Key Values: The models in the category use some labels to model certain context, and these key values could be keywords or task attributes. Uddin *et al*. proposed a CAT (A Context-Aware Trust) model to compare the similarity of contexts by using some keywords that could describe certain context to some extent (i.e. task A: My brother drives me to the airport, the keywords

may be {my brother, drive, car}; task B: My brother flies the plane, the keywords may be {my brother, fly, plane} [19]. Two out of three words are different, so it may be untrustworthy. Rehak *et al*. use some attributes of task to describe the context, and then utilize "Manhattan distance" to measure similarity. Clustering method is used to reduce computational complexity [10]. Caballero *et al*. define a formula using task attributes of context to calculate their similarity [2]. However, from the context modeling point of view, key values are easy to manage, but they lack of capabilities for sophisticated structure. Strang *et al*. provide a survey of different approaches to model context. Based on the comparison from several aspects, they conclude that key value-based model is inefficient for describing complex relations in contexts" [16]. For instance, in e-commerce environments, *Inkjet printer* and *digital projector* are two different products, but they have similarity to some extent, and both of them belong to the type of computer output device. Therefore, customers may trust the seller and buy a *Inkjet printer* provided that they know the seller has a good reputation on selling high quality *digital projector*.

2) Context Description Based on Hierarchical Ontology Structure: Strang *et al*. point out that ontology is a promising instrument to specify concepts and interrelations [16]. Most literature use the ontological structure to analyze contextual trust. Uschold *et al*. gave a good explanation on related conception of ontology [20]. Ontology is essentially a conceptual model, but could express the relationship between different concepts. The relationship of these concepts can be *"is_a"* and *"attribute"*. Toivonen *et al*. describe a trust determination process based on contextual information [18]. The authors use ontology structure of a network, over which some software components are downloaded. Suppose we need to download some software components from an unfamiliar node, the trust value of such an action can be calculated from its neighboring nodes. The influence of their neighboring nodes on trust evaluation depends on their "semantic distance" to the current node. Tavakolifard *et al*. propose an enhanced trust model that can be regarded as a complementary solution in comparison with the work in [18], which aims to find similar and relevant nodes in hierarchical structure [17].

In our paper, a contextual trust evaluation method is discussed in e-commerce and e-services environments, and it is based on transaction similarity calculation. Our method mainly focuses on product and service category and transaction amount properties, which play important roles in describing transaction context.

III. Contextual trust evaluation method based on transaction similarity

In order to obtain the trustworthiness of part B in a specific context, part A needs to take other parties' ratings on B in the same context into account. But when no trust information from others for the same context is available, A could use the trust information from the parties in the same hierarchy as reference, which share a common ancestor within the hierarchical ontology structure. Of course, the trust level of these parties will be discounted accordingly [17]. For example, *laser printer* and *photo printer* are both the subclass of *printer*, thus *printer* represents their common ancestor. Buyers can trust the seller and buy a *laser printer* from that seller, provided that they know the seller has a good reputation on selling *photo printers*. Therefore, how to establish a common hierarchical ontology structure to model context and how to measure "semantic similarly" of different contexts (e.g., *laser printers* and *photo printers*) within the whole hierarchical ontology structure are two important problems in context similarity (CS) calculation.

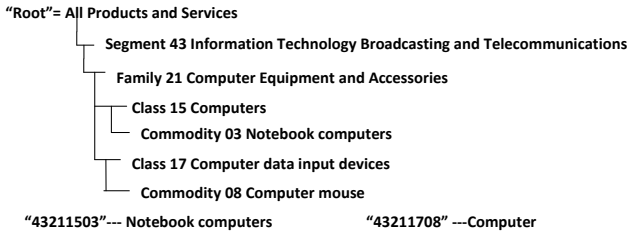

Figure 1. An example of coded commodities in UNSPSC

There are two important contextual properties of online transactions, i.e. product/service category and transaction amount, which greatly influence the trust evaluation. According to the proposed example above, product/service category is considered with a hierarchical structure to help buyers and sellers establish trust relationship. In addition to the hierarchy of product/service category, transaction amount is another transaction context hierarchy that influences trust evaluation. A buyer may not worry about the trust of a seller who has a good reputation on selling products/services with similar or higher price compared with what s/he wants to buy, while the buyer may worry if a seller wants to commit fraud, who always sells products/services at lower price, but begins to supply more expensive ones (*the value imbalance problem*).

A. Similarity Comparison Between Categories of Product and Service

1) Hierarchy of Product and Service Categories: There are existing some *Products and Services Categorization Standards (PSCS)* aim at grouping similar products/services, such as United Nations Standard Products and Services Code (UNSPSC)[2] and eCl@ss[3], which provide an industry-neutral ontology of product and service categories.

UNSPSC is a hierarchical classification, which enables "drill down" and "roll up" operations in analysis. There

[2]http://www.unspsc.org/

[3]www.eclass.de/

are three main design rules of UNSPSC. Firstly, products and services are grouped according to the dominating usage in the world market. Secondly, category titles are unambiguous and mutually exclusive. Thirdly, a product or service appears in only one category, and each category has one parent only. Each level in hierarchy contains a two-digit number, "segment" (the logical aggregation of families for analytical purposes), "Family" (a commonly recognized group of inter-related commodity categories), "Class" (a group of commodities sharing a common use or function), and "Commodity". Fig. 1 presents how "Notebook computers" and "Computer mouse" are coded in UNSPSC. The categories in UNSPSC respond to the marketplace that are grouped into 55 segments (top-level), which represent product/service in different domains, like "Electrical System 44", "Food and Beverage 50", "Healthcare Services 85". The segments are arranged in a logical sequence that reflects how value is progressively added to products (i.e. from "Raw materials" to "End Use Products" to "Services"). But family, class and commodity codes are arbitrary, and there are no logical sequence for them.

Some researchers pointed out that the major obstacle in using UNSPSC is that it is rather shallow and not descriptive on an attribute level. Hence, they built a new classification scheme eCl@ss, thereby replacing UNSPCS [15]. Similarly, eCl@ss also utilizes a hierarchical ontology structure and uses a two-digit number for each level. Fig. 2 presents part of ontology structure of "product" categories in eCl@ss version 6.2, and "service" categories in eCl@ss cover "Travel", "Logistics", "Finance and Insurance" etc. Its improvement lies in two aspects:

a) Some products and services in the last level are enriched with some keywords (these keywords make another layer) to assist buyers position their demands more accurately, and eCl@ss uses a standard set of attributes (i.e. *manufacturer name*, *product type*) to describe them. It is out of the scope of this paper to compare these two standards, but description at attribute level is an important reason for us to use eCl@ss, as these attributes can be utilized to extend leaf nodes in the hierarchy of product and service categories. In this way, similarity measure between products could be more accurate, this problem will be discussed in the next subsection. For example, *laptop* as the leaf node in eCl@ss in Fig. 2 can be extended with attributes, such as *manufacturer name* (*DELL*, *IBM* etc.), *product type*.

b) Products and services in eCl@ss are more functionally grouped, and they are subdivided for specific usage. For example, screw appears in several different categories in eCl@ss, as a screw to fix a painting on the wall is different from the screw a surgeon needs to fix broken bones in a human body [11].

Although eCl@ss is more suitable for our method, there are still some problems within this ontology structure. For example, the classifications in UNSPSC is strictly obey

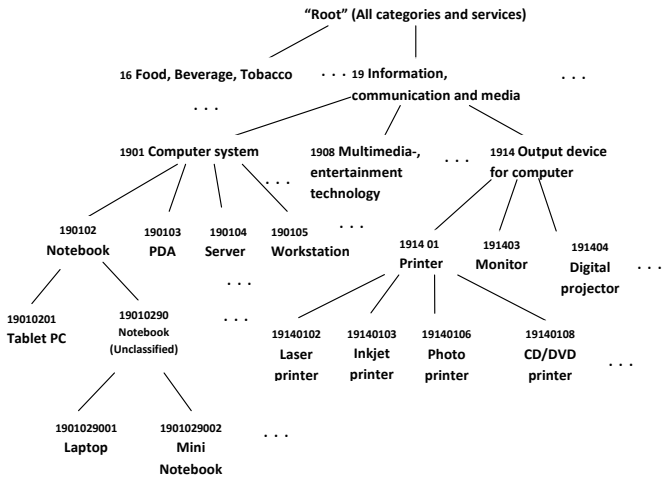

Figure 2. Ontology structure of segment *"Information, communication and media"* in eCl@ss version 6.2

inclusion relation (such as "is_a"), which is considered as a rule of ontology structure (see Fig. 1). While there are some fuzzy classifications existing in eCl@ss, for instance, "Notebook (Unclassified)" is a subclass of "Notebook" (see Fig. 2); "Food processing machinery (others)" is a subclass of "Food processing machinery", which names as *counter-intuitive classification* [15]. Therefore, some researchers proposed a challenge that borrows the standards of UNSPSC to improve eCl@ss [15]. However, they are currently the most important standards for product and service classification.

2) Similarity Measurement within Ontology of Product or Service Categories: To measure the similarity of two nodes within a hierarchical ontology structure, a crucial factor is the depth of the deepest common ancestor of the two nodes d. For instance, the deepest common ancestor of *laptop* and *PDA* is *computer system* (see Fig. 2), and its depth d is 2. An upper layer of the hierarchy represents more general classification with less similarity between them, while lower layers are more concrete with stronger similarity. Taking the above considerations into account, the similarity S_{pc} between two product categories should be a monotonically increasing function with respect to the deepest common ancestor of the two product/service categories. We use a hyperbolic tangent function to satisfy this trend. Thus, the similarity of two product/service categories c and c' could then be defined as the function of d:

$$S_{pc}(c, c') = \frac{e^{\alpha d(c,c')} - e^{-\alpha d(c,c')}}{e^{\alpha d(c,c')} + e^{-\alpha d(c,c')}} \quad (1)$$

where $\alpha > 0$ is a constant. Fig. 3 shows the influence of different α with respond to S_{pc}. From the ontology structure of product and service categories, we could easily find when $d \geq 3$ the categories of products and services are already quite close (e.g. *Tablet PC* and *laptop* have

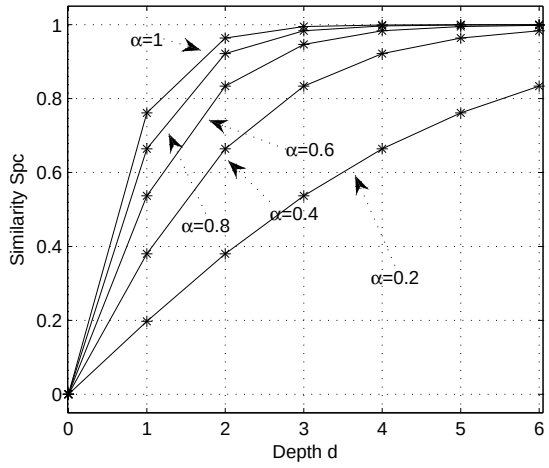

Figure 3. The influence of different α on similarity

Algorithm 1: Similarity of product and service categories

Data: c_{code} and c'_{code} are two codes of products or services in eCl@ss, p, m, d
Result: S_{pc}
begin
  compare each number in c_{code} and c'_{code};
  if c_{code} *and* c'_{code} *contain different number* **then**
    record position p;
    $m = p \ mod \ 2$;
    if m=1 **then** set $d = floor(p/2)$
    else set $d = (p/2) - 1$
  else c_{code} and c'_{code} are the same; return $S = 1$
  return $S_{pc}(c_{code}, c'_{code}) = \frac{e^{\alpha d} - e^{-\alpha d}}{e^{\alpha d} + e^{-\alpha d}}$
end

strong similarity, their common ancestor is *Notebook*), while $d \leq 2$ the different product and service categories have weak similarity based on human common sense (e.g. *Printer* and *Digital projector*, their common ancestor is *Output device for computers*). Therefore, according to the curves plotted in Fig. 3, we set an parameter $\alpha = 0.4$. Algorithm 1 presents how to find the deepest common ancestor of two products or services c and c' in a category hierarchy, and then return the depth d of their ancestor. Finally, similarity of different categories is calculated. Input data are two codes of products or services in eCl@ss. The results of similarly measure, for example, S_{pc}(*Tablet PC*, *Laptop*) is approximately 0.83, S_{pc}(*Monitor*, *Digital projector*) is 0.66, S_{pc}(*Laptop*, *Inkjet printer*) is 0.38, S_{pc}(*PDA*, *food*) is 0 (the similarity is 0, when two products belong to different segments as plotted in Fig. 2).

Additionally, S_{pc} also can represent the similarity of product values in the world market. For example, *Tablet PC* and *laptop* share the same upper-level category (i.e. Notebook in the third level in Fig. 2), S_{pc} of them is as high as 0.83, and their value in world market is also quite close. When the value of similarity S_{pc} is decreasing, which means the deepest common node of two different categories products/services in a higher level, their value in world market will no longer be close. For instance, although *Laptop* and *printer* share the same upper-level general classification within the category ontology structure, namely, they all belong to *information, communication and technology products* (see Fig. 2), but some low value *printers* are worth less than \$100, while the price of *laptops* are always higher than \$500. According to the analysis above, we could say when the value of S_{pc} is lower than certain degree (i.e. $S_{pc} \leq 0.8$), apart from category similarity, transaction amount similarity also plays an important role in trust evaluation, as the fraud may happen (e.g. *the value imbalance problem*).

B. Similarity Comparison Between Transaction Amount

Similarity comparison in transaction amount is important from the risk point of view. Different situations can be further analyzed in two cases as follows.

1. The past transaction amounts are the same or larger than that of the new one. For instance, consider a seller S who has a lot transactions with some buyers, wherein each transaction amount is basically around $\$1K$-$\$2K$. Assume the service is good each time. Let B denote the set of corresponding buyers. Now S is going to sell a product for \$100. If a buyer b' knows that the trust ratings over S done by other buyers in B are quite positive, b' may not worry about the trust of the new transaction as the product to buy is cheaper.

2. The past transaction amounts are less than that of the new one. In this case, the past transaction cannot be taken as a direct reference because of risk. For instance, a seller may be always honest when selling cheap items (say \$100). However if the new product is quite expensive (say $\$5K$ or more), a fraud is more likely to happen (i.e. *the value imbalance problem*). Due to this reason, after having collected the ratings of past transactions with lower transaction amounts, a factor should be determined to discount these ratings.

To summarize the above analysis, some similarity comparison principles can be listed as follows:

Principle 1**:** The past transaction amounts can be taken into account to calculate the similarity of a new transaction provided that the past transaction amount is much less than the new one. The larger the difference is, the less similarity will be.

Principle 2**:** The past transaction amounts will have minor impact on a new transaction provided that the past transaction amounts are similar or much higher than the new one. In this case, the similarity between the past transaction amounts and the new transaction amount is 1.

1) Create Transaction Amount Hierarchy: To calculate the similarity of transaction amount, we firstly create a tree structure for transaction amount. From buyers' point of view,

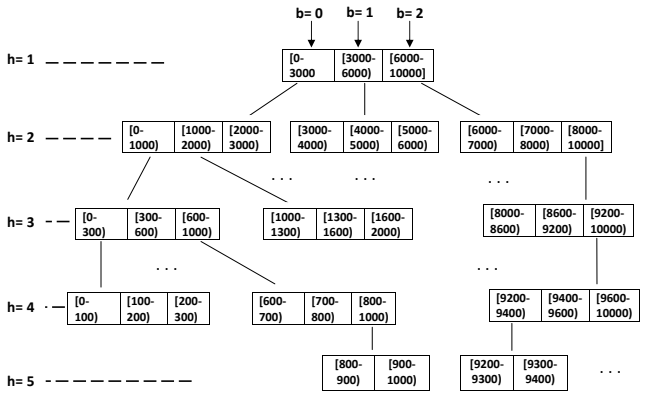

Figure 4. A example of transaction amount tree

different ranges of transaction amounts mean different level of risk that a buyer needs to take in a new transaction.

To build this transaction amount tree, we define two parameters N and R. Similar to N-ary tree, the value of N specifies the number of children per non-leaf node, and R is the range of transaction amount at the leaf nodes of the tree. In some E-commerce environment websites, the transaction amount is usually limited within large transaction amount (i.e. \$10000). The transaction amount tree is plotted in Fig. 4 ($R \leq 100, N = 3$).

2) The Similarity Factor: Our method defines a similarity factor S_{ta} to measure the influence of transaction amount on trust level; its value is determined by two parameters: the depth of the deepest common ancestor of the two leaf nodes h (higher level means less similarity and more risk). As each non-leaf node has N children, and different sibling nodes have different ranges of transaction amount. Thus, longer distance between sibling nodes also means more risk and less similarity, the distance between them is b. For example, in Fig. 4, the values of h and b corresponding to \$50 and \$250 are 4 and 2, respectively. In transaction amount tree, the interval of transaction amount is increased/decreased with h and b, hence similar to category similarity S_{pc}, the similarity S_{ta} of two transaction amounts a and a' is defined as follows:

$$S_{ta}(a, a') = \frac{e^{\beta h(a,a')} - e^{-\beta h(a,a')}}{e^{\beta h(a,a')} + e^{-\beta h(a,a')}} * [1 - \gamma \sin(\frac{\pi b(a, a')}{2N})] \quad (2)$$

where N is the number of children per non-leaf node, $\gamma, \beta \in (0, 1)$ are two parameters scaling the similarity factor, and using two parameters $\gamma = 0.2$ and $\beta = 0.4$ in our method.

The similarity factor is calculated in Algorithm 2. Input data Avg is the average transaction amounts of all the past transactions of a seller in Eq. 3. Suppose that given n data objects or points to represent all previous transaction amounts of a seller form a set $\{x_1, x_2...x_n\}$, data L represents the forthcoming transaction amount. In Algorithm 2,

Algorithm 2: Similarity factor S_{ta}

Data: two nodes Avg and L, array r, s, set $\{x_1, x_2...x_n\}$, $difference$, h, b
Result: S_{ta}

```
set Avg = (Σ_{i=1}^{n} x_i)/n, difference = 0, h = 0, b = 0;
begin
    both Avg and L scan the tree to position the leaf node they belong to from first level "root";
    while node ≠ "leafnode" do
        find the branch node that Avg and L belong to from N branches of non-leaf nodes;
        add their positions to the array r and s, respectively;
        Avg and L continue to scan the children of branch nodes;
    while difference ≠ 0 do
        compare each number in two arrays r and s;
        if r and s contain different number then
            record position h;
            b is difference between r and s in position h;
            difference =0;
        else d + 1
    if b < 0 return S_ta(Avg, L) = (e^{βh} - e^{-βh})/(e^{βh} + e^{-βh}) * [1 - γ sin(π(-b)/2N)]
    else return S_ta = 1
end
```

we use BSF (Breadth-First-Search) to locate the leaf nodes of two transaction amounts. Two arrays r and s record the path from "root" to leaf node to trace the nodes Avg and L. For example, to locate \$50 in the transaction amount tree, the path code $r = 1111$, and to locate \$250, the path code $s = 1113$. The algorithm compares two path codes r and s. They are different in the fourth number, and the difference is 2 ($h = 4$, $b = 2$).

$$Avg = \frac{\sum_{i=1}^{n} x_i}{n} \quad (3)$$

Suppose that a seller begins to supply different types of products/serivces with transaction amount \$250, but the past transaction amounts of this seller are always around \$50, and then the similarity of transaction amount is $S_{ta} = 0.76$. It is true that the less risk a buyer will take, the more trustworthiness the seller will be. When the transaction similarity S_{ta} decreases to a certain low level, the buyer can suspect the seller may commit frauds.

C. Contextual Trust Evaluation Method

In some real E-commerce systems, the trust rating scores of a seller given by raters are represented by a series of fixed numbers. For instance, the ratings at eBay, are in the set $\{-1, 0, 1\}$. At Epinions[4], each rating is an integer in $\{1, 2, 3, 4, 5\}$. These ratings can be used for trust evaluation. Since ratings as numerical values in $[0, 1]$ are more suitable for trust evaluation [6], we normalized the five-level scale at Epinions. Fig. 5 presents 100 ratings of a seller. The raters corresponding comment set can be labeled as $\{terrible, poor, medium, good, excellent\}$ equivalent to $\{0, 0.25, 0.5, 0.75, 1\}$. In this paper, we propose a two-phase

[4]http://www.epinions.com/

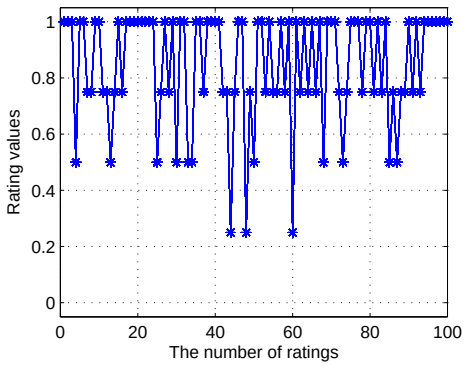

Figure 5. Ratings from Epinions

contextual trust evaluation method. On the one hand, the trust level of forthcoming transaction will be inferred, on the other hand, some potentially malicious transactions can be identified and prevented.

Trust Data Representation In order to evaluate the trust level of a forthcoming transaction, we assume the following trust data structure.

$$TR =< S; B; R^{(i)}_{B\to S} > \tag{4}$$

1. TR is the transaction between seller S and buyer set B;

2. $R^{(i)}_{B\to S} \in (0, 1)$ is the rating given by B;

Given n ratings $(R^{(1)}, R^{(2)}, ..., R^{(n)})$, trust value of seller S is

$$T_s^{(R^{(1)},R^{(2)},...,R^{(n)})} = \frac{\sum_{i=1}^{n} R^{(i)}}{n} \tag{5}$$

Phase 1: If a seller sells different types of products/services, the previous ratings over this seller will be discounted, and this discount is greatly associated with similarity of product/service category. Trust value after comparing products category similarity is computed in Eq. (6).

$$T_{pc} = \frac{\sum_{i=1}^{n} (1-\omega) * S_{pc} * R^{(i)} + \omega * R^{(i)}}{n} \tag{6}$$

where we use heuristic parameter ω to weight the influence of previous ratings.

Phase 2: Trust discount after comparing transaction amount similarity, and final trust value of this seller can be computed as follows:

$$T_{final} = S_{ta} * T_{pc} \tag{7}$$

IV. An Example

In this section, we use an example to illustrate how our proposed contextual transaction trust evaluation method works.

Example: Consider a scenario that a buyer plans to buy a certain model of DELL laptop. Four sellers S_1, S_2, S_3 and S_4

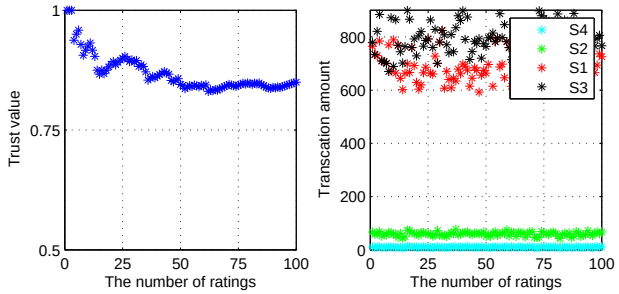

Figure 6. (a) Initial same trust level of four retailers. (b) Previous transaction amount of four retailers.

provide this laptop with an attractive price of around \$900. Assume four sellers sold different products before and just start to sell laptops, and the products that they sold before are S_1:Tablet PC, S_2:low value inkjet printers, S_3:expensive handbags and S_4:food. We also assume they earned the same ratings in past transactions as plotted in Fig. 5, and their previous transaction amounts are plotted in Fig. 6 (b).

Their initial trust evaluation results are the same around 0.88 as plotted in Fig. 6 (a). However, based on the proposed method, we can infer that the trust value $T_{pc_{S_1}}$ is approximately 0.81, $T_{pc_{S_2}}$ is 0.61 after comparing category similarity (S_{pc}) in Phase 1. For each of S_3 and S_4, the domain of laptop is different from the domain of products they sold before (i.e. different segments in category classification in Fig. 2), and their trust values are as low as 0.44 ($\omega = 0.5$). The trust evaluation results of four sellers in Phase 1 are shown in Fig. 7.

Fig. 8 plots the fluctuation of trust level over these four sellers when taking transaction amount similarity into account after Phase 2. According to Algorithm 2, we can obtain their final trust values $T_{final_{S_1}} = 0.81$, $T_{final_{S_2}} = 0.42$, $T_{final_{S_3}} = 0.38$, and $T_{final_{S_4}} = 0.30$.

Analysis: The similarity S_{pc} of *Tablet PC* and *laptop* is as high as 0.83 ($S_{pc} \geq 0.8$), and their values in world market are also quite close. Thus, for S_1, the similarity factor S_{ta} needs not to be used to evaluate the changes in trust value over the sellers in phase two. The previous ratings over this seller could be used directly to help buyers make trust decision.

When the value of S_{pc} is decreasing, the value of two in world market will no longer be close to each other. In such a case, besides category similarity S_{pc}, transaction amount similarity S_{ta} is also important to quantify how much risk a buyer may have to take. If it is a high risk, it may make buyers no longer trust this seller again. For example, the trust level of S_2 drops from 0.88 to only 0.42.

For S_3 and S_4, they just begin to sell laptop which is different from the domain of products/services they sold before. Therefore, their previous ratings will not be taken as the major reference. Furthermore, if the sellers' good reputation is earned by selling products/services at low price

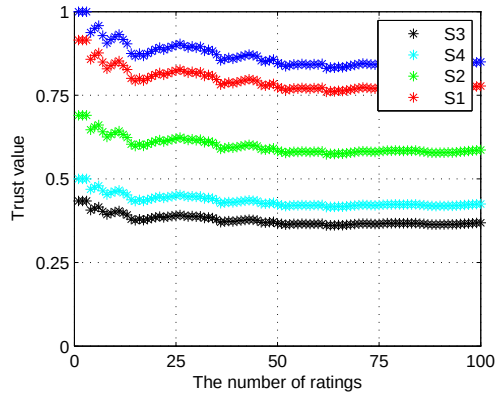

Figure 7. Trust evaluation after Phase 1

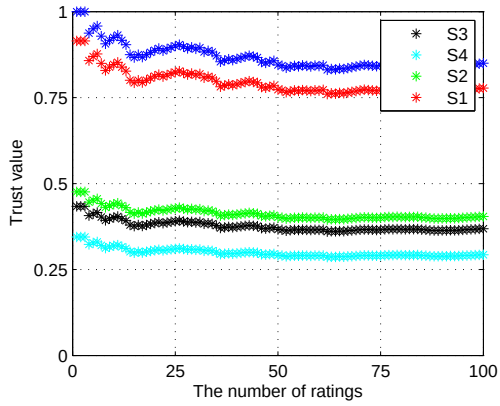

Figure 8. Trust evaluation after Phase 2

in different categories, the buyers have good reasons to suspect the sellers may commit frauds (i.e. S_4).

V. Conclusion

In this paper, we have proposed a contextual trust evaluation method taking transaction similarity into consideration. From calculation category similarity point of view, when a seller sells products/services in different categories, our method can infer the trust level of this seller for a forthcoming transaction. Furthermore, in order to prevent that the sellers may earn good reputation via selling cheap products, and start to cheat buyers by selling expensive products/services (i.e. the *value imbalance problem*), we introduce transaction amount similarity S_{ta} to re-evaluate the trust level of sellers. When the value of similarity S_{pc} is high enough (say $S_{pc} \geq 0.8$), previous trust ratings of this seller can be used directly, as two different products/services also have similar market values. But when S_{pc} is small, apart from category similarity, at this time, taking transaction amount similarity S_{ta} into account is more important to measure whether a seller wants to commit fraud.

References

[1] A. Abdul-Rahman and S. Hailes. Supporting trust in virtual communities. In *33rd Hawaii International Conference on System Sciences HICSS '2000*, volume 1, page 9.

[2] A. Caballero, J. Botia, and A. Gomez-Skarmeta. On the behaviour of the trsim model for trust and reputation. In *ACM MATES'2007*, pages 13–24.

[3] N. Griffiths. Task delegation using experience-based multi-dimensional trust. In *AAMAS'05*, pages 489–496.

[4] S. Holtmanns and Z. Yan. Context-aware adaptive trust. In *Developing Ambient Intelligence' 2006*, pages 137–146.

[5] R. Kerr and R. Cohen. Modelling trust using transactional, numerical units. In *ACM PST'2006*.

[6] L. Li and Y. Wang. Subjective trust inference in composite services. In *AAAI' 2010*, pages 1377–1384.

[7] S. Marsh. Formalising trust as a computational concept. *PhD Thesis*, 1994.

[8] D. H. McKnight and N. L. Chervany. The meanings of trust. *Technical Report*, pages 94–04, 1996.

[9] L. Mui. Computational models of trust and reputation: Agents, evolutionary games, and social networks. *PhD Thesis*, 2003.

[10] M. Rehak, M. Gregor, M. Pechoucek, and J. M. Bradshaw. Representing context for multiagent trust modeling. In *IEEE/WIC/ACM International Conference on Intelligent Agent Technology. IAT '2006*, pages 737–746.

[11] P. J. A. Reusch and L. E. G. Moreno. Introduction of entropy functions to improve the classification of commodities in e-commerce. In *IEEE International Workshop on IDAACS'2009*, pages 575–580.

[12] B. Rietjens. Trust and reputation on ebay: Towards a legal framework for feedback intermediaries. *Information and Communications Technology Law*, 15(1):55 – 78, 2006.

[13] J. Sabater and C. Sierra. Regret: Reputation in gregarious societies. In *ACM AGENTS '2001*, pages 194–195.

[14] J. Samek and F. Zboril. Hierarchical model of trust in contexts. In *In Networked Digital Technologies. Communications in Computer and Information Science (CCIS)'2010*, pages 356–365.

[15] E. Schulten, H. Akkermans, N. Guarino, G. Botquin, N. Lopes, M. Dorr, and N. Sadeh. Call for participants: The e-commerce product classification challenge. *IEEE Intelligent Systems*, 16(4):86–93, 2001.

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

[17] M. Tavakolifard, S. J. Knapskog, and P. Herrmann. Trust transferability among similar contexts. In *ACM Q2SWinet'2008*, pages 91–97.

[18] S. Toivonen, G. Lenzini, and I. Uusitalo. Context-aware trust evaluation functions for dynamic reconfigurable systems. In *Models of Trust for the Web Workshop,WMT'2006*, volume 190.

[19] M. Uddin, M. Zulkernine, and S. Ahamed. Cat: A context-aware trust model for open and dynamic systems. In *ACM Symposium on Applied Computing, SAC'2008*, pages 2024–2029.

[20] M. Uschold and M. Gruninger. Ontologies: Principles, methods, and applications. *Knowledge Engineering Review*, 11(2):93–155, 1996.

[21] Y. Wang and E. P. Lim. The evaluation of situational transaction trust in e-service environments. In *ICEBE'2008*, pages 265–272.

[22] Y. Wang and K. J. Lin. Reputation-oriented trustworthy computing in e-commerce environments. *IEEE Internet Computing*, 12(4):55 – 59, 2008.