Journal of Data Intelligence, Vol. 2, No.2 (2021) 101-115

FUZZY TRUST MODELING FOR PERVASIVE COMPUTING APPLICATIONS

KOSTAS KOLOMVATSOS, MARIA KALOUDA, PANAGIOTA PAPADOPOULOU, STATHES HADJIEFTYMIADES

Department of Informatics and Telecommunications,

National and Kapodistrian University of Athens

{kostasks, std04014, peggy, shadj}@di.uoa.gr

Pervasive computing applications involve the interaction between autonomous entities for performing complex tasks and producing knowledge. Autonomous entities can interact to exchange data and knowledge to fulfil applications requirements. Intelligent Agents (IAs) 'activated' in various devices offer a lot of advantages when representing such entities due to their autonomous nature that enables them to perform the desired tasks in a distributed way. However, in such open and dynamic environments, IAs should be based on an efficient mechanism for trusting unknown entities when exchanging data. The trust level of an entity should be automatically calculated based on an efficient methodology. Each entity is uncertain for the characteristics and the intentions of the others. Fuzzy Logic (FL) seems to be the appropriate tool for handling such kind of uncertainty. In this paper, we present a model for trust calculation under the principles of FL. Our scheme takes into consideration the social dimension of trust as well as personal experiences of entities before they decide interactions with an IA. The proposed model is a two-level system involving three FL sub-systems to calculate (a) the social trust (based on experiences retrieved by the community), (b) the individual trust (based on personal experiences) and (c) the final trust. We present our results by evaluating the proposed system compared to other models and reveal its significance.

Key words: Pervasive Computing, Autonomous Entities, Trust, Reputation, Fuzzy Logic

1 Introduction

The rapid evolution of pervasive computing sets new challenges in the research community about the development of new services and applications. The combination of wireless technologies (e.g. Wireless Sensors Networks) and the Internet accompanied by the respective hardware allows for numerous nodes to be interconnected. Pervasive computing involves the adoption of numerous devices embedded into everyday objects for supporting intelligent applications. The transition from closed networks to interconnected autonomous nodes interacting with their environment and performing simple processing tasks should be enhanced by intelligent applications increasing the quality of services that end users enjoy.

Pervasive computing applications usually involve the interaction between autonomous nodes to exchange data and knowledge, creating a complex architecture. Such architectures can assist in deriving knowledge necessary to support complex services and applications. The automated knowledge discovery and data exchange can be realized by Intelligent Agents (IAs) having the form of software or hardware components capable of acting autonomously to achieve goals defined by their owners. However, IAs should retrieve data and knowledge and rely upon trusted entities (i.e. other

IAs) to ensure a reliable data exchange and efficiently support pervasive applications. Usually, IAs take into consideration the reputation and trust levels of other Ias to start and conclude an interaction. However, the calculation of the trust level of an external entity is a very difficult task.

Trust management has long been a significant domain in Computer Science and refers to various aspects of entities behaviour, in areas such as e-commerce. The meaning of the trust concept varies depending on the context [2]. Trust can be seen as the extent to which one entity intends to depend on somebody else in a given situation [18]. When focusing on interactions between autonomous entities, one can easily detect the uncertainty behind any decision for concluding these interactions. Such an uncertainty refers in the intended behaviour that an entity may possibly exhibit rendering it as suitable to be trusted or not. For handling this uncertainty, we propose a Fuzzy Logic (FL) based system for estimating the trust level of an IA.

We define an efficient modeling process that seeks to imitate human behavior. IAs aim is to interact only with those entities having a high trust value. The system is based on: a) the social aspect of trust, b) the individual experiences of each IA and c) their combination. The significance of the proposed model is that it combines both social and individual trust values in an efficient way. The proposed model employs a distributed approach as IAs can calculate the trust level in an autonomous manner. The proposed model extends previous research in two aspects: (i) We do not deal with binary ratings or specific values like in Ebay (-1, 0 or 1) [12], [24]. Binary ratings are considered insufficient to capture various degrees of judgment [1]; (ii) In previous models [12], [13], [15], trust is based only on one value, the final rate, i.e. the final trust value. In the proposed model, trust is defined and estimated by a number of parameters, thus every referrer rates the examinee for these parameters.

The remainder of this paper is organized as follows: Section 2 reports prior work while Section 3 gives the necessary description of our scenario. Section 4 is devoted to the description of our model analyzing its three sub-systems for calculating the social, the individual and the final trust. In Section 5, we discuss our results while Section 6 concludes our paper.

2 Related Work

The authors in [7] define the notion of trust and describe simple models for trust calculation. Trust can have a cognitive and a mathematical aspect, which involve the underlying beliefs concerning trust as well as equations for trust extraction. The Fire system is described in [10], a trust and reputation model integrating various information sources to calculate IA performance, based on interaction trust, role-based trust, witness reputation and certified reputation. In [14], the authors provide a detailed overview of reputation and trust models highlighting their importance to open environments. A categorization of trust is presented in [20]. Decentralized and centralized trust is reviewed in [24], presenting a model with Bayesian networks combining different trust aspects, applied for a file sharing peer-to-peer application.

In [1] a trust establishment model is described going beyond trust evaluation to outline actions to guide trusters. The model relies on a multicriteria approach for measuring and analysing trusters needs and evaluates the satisfaction level of trusters based on their values and expressed preferences. The authors of [2] propose a framework for trust calculation based on the assumption that the more values agents share, the more they should trust one another. The model relies on agents' past behavior to

conclude the final trust taking into account if agents trust cautiously or boldly, and if they depend on others in carrying out a task. According to [26], IAs from the same system should be evaluated differently from agents in different multi-agent systems. The trust level is affected by the platform they are activated in. In [6] a model of trust that incorporates competence and integrity is proposed. The threshold for trustworthiness in a particular context is viewed as a function of agents' relationship with the truster and potential impacts of decisions. In [5], the human-agent collaboration is studied with current approaches to measure trust and how they can be inadequate in a real time setting, critical to know the user's trust in the agent.

FL has been widely used for evaluating trust. An FL based system for trust and reputation calculation is presented in [2], which is actually a Fuzzy extension of the Beta reputation model in [12]. Two fuzzy subsets are proposed, namely 'satisfied' and 'unsatisfied'. Based on the combination of the two fuzzy sets, the authors present mathematical formulations for calculating the agreement level between two partners. In [8], trust calculation between cooperative IAs is studied. FL is used for representing trust, allowing IAs to handle uncertainty. The authors present a mechanism for agents to make use of distrust. Distrust has not only a negative meaning, the opposite of trust, but represents the belief that an IA will act against the goals of another. Membership functions and the FL rule base are studied. In [9] a proposed model for trust calculation based on FL is presented, taking into consideration different trust sources aiming to minimize wrong evaluations. The final trust is based on a weighted fuzzy calculation. In [15], a comparison between fuzzy aggregation models and existing methods for trust evaluation is presented. FL is used to build the final trust level based on a number of values that should be aggregated. The results show superiority of the proposed FL algorithm. The adaptation of a bio-inspired trust model to deal with linguistic fuzzy labels seems to be more efficient and closer to the human way of thinking [17]. Linguistic fuzzy sets represent the satisfaction level of a client. The model calculates the final trust value in five steps.

In [19], the authors describe a model that attempts to identify the information customers expect to find on vendors websites in order to increase their trust and, thus, the likelihood of a successful transaction. Fuzzy reasoning can handle imprecise data and uncertainty when measuring the trust index of each vendor. Trust for mobile IAs is studied in [22]. Customers can collect feedback using IAs and, thus, build trust. Once a customer performs a transaction with a provider the feedback is received and, thus, can be taken into consideration in the trust calculation.

A reputation-based model for trust management in a semantic P2P Grid is proposed in [11], using fuzzy theory for computing peer trust level. The research work combines FL and a reputation model in a reputation collection and computation system, to infer trust. Based on network structure and storage of reputation information, semantic is used a fuzzy parameter for clustering the grid environment, aiming to increase peer trust. In [23] the authors present a model for reputation-based trust, incorporating FL. They extend their previous work on reputation-based trust with the introduction of fuzzy subsystems for estimating importance of transaction, the decision to trust and the interaction result.

3 Preliminaries

3.1 Trust

Trust is a key concept in various contexts, including transactions between IAs. In order to interact with their peers, IAs should be able to rely on an efficient mechanism enabling the estimation of trust of other entities. Hence, trust can be seen as part of a directional relationship of unknown entities that try to interact in order to exchange data and services. This term is presented in [21], discerning hard and soft security mechanisms. Hard tools are authentication or cryptography while soft tools involve social control issues (i.e. trust and reputation). In the relevant literature, the term 'trust' could be met with a variety of meanings. It is important to separate trust into one's ability to propose to another called a referral trust and the real trust known as a functional trust. Based on [8] trust can be defined as:

Trust (T) represents an agent's individual assessment of the reliability of another to accomplish a task

Trust can be interpreted as an entity's credibility regardless of any actual commitment which may indicate a subjective judgment that a entity A expects that another entity B will complete a specific action on which A's interest depends. Hence, trust is primarily defined as the trustor's assessment of the trustee's credibility (for instance, expressed in a probabilistic manner) in the context of trustee dependence. Trust is a complex, dynamic and context-specific phenomenon, largely based on beliefs that an entity has for another [18]. Such beliefs are largely subjective and of unclear origin. This means that an IA may be reliable only for a set of IAs and not for all of them. The level of trust is also dependent on the context. For example, an IA may be trustworthy for providing information but not for selling products. Furthermore, trust is dynamic. An IA may consider another entity as reliable in a specific time point but its opinion may change by the behaviour of the target entity. In general, trust can be considered as a function of the following parameters: the beliefs of the examiner, the reputation of the examinee, previous trust values and the context.

3.2 Reputation

Reputation is a concept representing a belief which has a social aspect and has been the topic of study in various fields, including IA systems, in conjunction with trust. Mainly, reputation reflects the opinion that the society has for a specific entity. Reputation can be seen as an overall measure of and entity's reliability based on recommendations or ratings from other members of a group. This means that the measure of trust for each entity active in a system can be reported in combination with the existing recommendations for the specific entity and past individual experiences. In order to avoid loops, it is necessary that recommendations are based on 'one hop' experiences only. In any case, reputation can characterize a group or an individual entity. The reputation of a group can, for instance, be modelled as the mean of the reputation values of all group members or as the way the whole group is perceived by other groups in the community. Every entity in the group may inherit an a priori reputation degree based on the reputation of the group. Based on [8] reputation can be defined as follows:

Reputation (R) is a social concept corresponding to a group assessment of the reliability of an entity to accomplish a task

The concepts of reputation and trust are closely related but different. The main differences between trust and reputation are: (a) Usually, trust is a score that reflects the subjective view of an entity for another entity whereas reputation is a score that reflects the view of the community for an

entity; (b) In trust systems, the transitivity aspect is considered explicitly, while in reputation systems, it is seen implicitly [25].

In our case, the key issue is the trust's dynamic nature. Trust evolves over time as entities cooperate with others. For this reason, it is critical to define a trust update process. Especially, in open environments like those in pervasive computing applications, where goals, beliefs and intentions of each IA change continually, there is a need for dynamic adaptation of trust. For this purpose, we utilize FL for handling such kind of uncertainty. In FL models trust and reputation are described with linguistic fuzzy values. Fuzzy inference is adopted in order to determine trust. Personal experience typically carries more weight than second-hand trust referrals or reputation, but in the absence of personal experience, trust often has to be based on referrals from others.

3.3 Referrals

Trust estimation can depend on referrals made by the society members for an entity based on their interactions with it. Based on [12], referral can be defined as:

Referral is the individual assessment of a third entity for the trust level of another

Apart from their own experience with an entity or in the lack of interaction history with it, IAs can be based on other IAs assessments to determine an entity's trust level. Usually, there is a central authority responsible for handling referrals. Every IA that wants to calculate the trust level of an entity relies on this authority and retrieves referrals for the entity, with which the IA can calculate the social trust value that reflects the society's opinion for the specific entity.

In our case, a key issue is trust's dynamic nature. Trust evolves over time as entities cooperate with others. Thus, it is critical to define a trust update process. Especially, in open environments like those in pervasive computing, where goals, beliefs and intentions of IAs continually change, there is a need for dynamic adaptation of trust. For this purpose, we use FL for handling this uncertainty. In FL models trust and reputation are described with linguistic fuzzy values. Fuzzy inference is adopted to determine trust. Personal experience typically has more weight than second-hand trust referrals or reputation, but in absence of personal experience, trust often has to be based on referrals from others [12].

4 The Fuzzy Trust Model

4.1 High Level Architecture

The proposed system assumes an environment where multiple entities may interact to perform some actions according to a pre-defined plan. These actions may also require to realize interactions with other entities active in the group. For instance, if we focus on an electronic market, we can easily discern the required interactions between buyers and sellers before a purchase is concluded. Other examples involve the tasks or data sharing between entities when acting in environments like the Internet of Things (IoT). Every entity in the system has a reputation which is calculated upon past behavior and the performed interactions in the community. We also consider a central authority responsible to manage the storage and access on the provided referrals. This central authority holds all the necessary data regarding the realized interactions that take place within the community and data related to the reputation of each entity. Additionally, every entity keeps locally a knowledge base to manage past individual interactions and their outcome. The evaluation of the outcome of an interaction is considered a social as well as an

individual element as it refers not only on the 'personal' experience and opinion of an entity but an indication for the remaining entities in the group. We have to notice that an evaluation may be different if it is for individual or it is extracted by the community.

Trust can be calculated by multiple parameters depicting the behavior of an entity. Our model focuses on detecting if an entity can offer timely a specific service with a desirable quality. The reason is that we want to have IAs collaborating to support pervasive computing applications, thus, we want to have real time quality responses. We propose that the trust value is calculated based on two aspects (a) the quality of the offered data/service; (b) the time required for other IAs to produce a response to IA's requests. These parameters can be easily extended by many more. Referrals concern realizations of the aforementioned parameters and represent the experience of IAs retrieved by interactions with the specific IA that offers the data/service. Simultaneously, IAs store information about the reputation of others and use each referral in combination with the reputation of the referrer. The aim is to be protected by entities that provide false referrals. For instance, if a referral is made by an IA with high reputation, it can affect the final social trust much more than a referral made by an entity with a low reputation. Every referral consists of two values in [-1,1], one for each attribute (quality, communication). We consider that each entity after completing an interaction with another entity performs a rating for each parameter of the interaction as exposed above storing it in a central repository. All entities have access to this repository of referrals when desired. Every referral is time stamped, thus, the temporal aspect in the management of the outcome of interactions is secured. Obviously, the older the referral, the less it contributes to the overall degree of trust as we incorporate a mechanism for minimizing the effect of old referrals in the final result.

The proposed system has three sub-systems (Fig 1) calculating respectively: **(a)** the social trust value, based on referrals by others; **(b)** the individual trust value, based on IA's personal experiences and **(c)** the weights for the social and the individual trust values to have the final one.

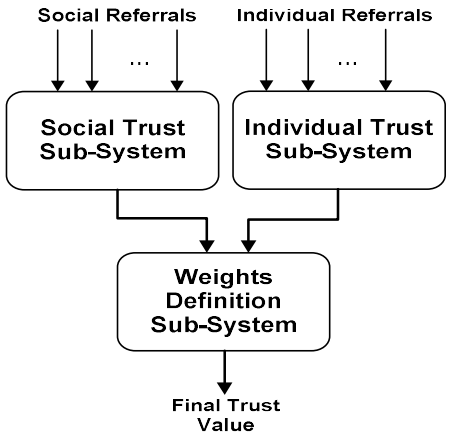

Fig 1. The proposed system

4.2 The Social Trust Subsystem

We provide a FL scheme for the social aspect of trust. An IA completing an interaction with another grades it for the two major system parameters: quality and speed of interaction. Each parameter is graded

with a real number in [-1,1]. Each referral has a timestamp and is stored by a central authority so as to be available to all entities. The degree to which a referral contributes to social trust is inversely proportional to the age of the referral. The older the referral, the less it contributes to the final social trust. Each referral has an expiration point, over which it is no longer taken into account. Another factor that influences social trust is the referrer reputation. Every IA is characterized by a degree of reputation, represented by a real number in [0,1]. The degree to which a referral contributes to social trust is proportional to the reputation of the referrer. Referrals coming from IAs with a high reputation (close to 1) affect social trust more. We propose an FL model with referrals as inputs and social trust as the output. If no referrals are available for an IA, social trust is set equal to 0 (a neutral value).

The social trust value is calculated by

$$\mathrm{ST}_{s_i} = \frac{\sum_{j=1}^{n}(T_j \cdot R \cdot F)}{n} \quad (1)$$

where ST_{s_i} is the social trust s_i, and n is the number of referrals made by community members for s_i. T_j is the weight of each referral according to its age, defined as

$$\mathrm{T_j} = \max(1 - \mathrm{e}^{\frac{\mathrm{d_j} - \mathrm{d_{max}}}{\mathrm{sm}}}, 0) \quad (2)$$

where d_j is the difference of days between the referral timestamp and the time that the IA calculates the trust, d_{max} is a maximum value (in days) and sm is a 'smoothing' factor. sm affects the strategy that the IA follows for realizing the weight of the referral. The higher the sm is, the more 'strict' the IA becomes. When sm is high, even for a low d_j (i.e., the referral is recent), T_j is very low (close to 0) and the referral has limited contribution to the final value. Moreover, over the d_{max}, a referral is characterized as obsolete and T_j is 0, i.e. it does not contribute to the calculation process. Parameter R refers to the reputation of the entity making the referral and is defined as

$$\mathrm{R} = (\mathrm{a} \cdot \mathrm{e_m} + (1 - \mathrm{a}) \cdot \frac{\mathrm{R_m}}{\sum_{\mathrm{i}=1}^{\mathrm{E}} \mathrm{R_i}}) \quad (3)$$

The reputation of the referrer is calculated from the opinion of the IA about the referrer and the reputation of the other IAs making referrals about the referrer. We consider that there is a mechanism handling reputation values, which is beyond the scope of this paper. Parameter e_m is a number in [0,1] representing the personal experience of the IA with the entity making the referral (m is the index of the specific entity). The e_m is based on ratings that the IA gives to any entity having an interaction with. The closer to 1 the e_m is, the more the IA trusts the entity making the referral. Parameter a is constant (defined by the developer) and represents the weight of parameter e_m, i.e., the weight of the IA's opinion. The right part of the sum in the above equation refers to the reputation of the referrer as a percentage of the sum of the reputation values of all the entities that made referrals for the specific IA (E is the number of entities making referrals for the referrer). We involve in the calculation process the opinion of the IA community for the entity. The higher the number of entities, the lower this factor becomes, even for large reputation values. With parameter a, the IA increases or decreases the weight of her opinion or that of the community. If a=1 the referrer reputation is based only on the IA's experience with it while if a=0 the reputation is based only on the community opinion. Last, parameter F is the result of the FL sub-system, defined as

$$\mathrm{F} = \frac{\sum_{\mathrm{k}=1}^{\mathrm{N}} \mu_{\mathrm{m}}(\mathrm{u_k}) \cdot \mathrm{u_k}}{\sum_{\mathrm{k}=1}^{\mathrm{N}} \mu_{\mathrm{m}}(\mathrm{u_k})} \quad (4)$$

The fuzzification process involves triangular membership functions ($\mu_m(u_k)$) applied for the two input variables, i.e. the IA's individual opinion and the community's opinion about the referrer. Each membership variable takes values in [-1,1] for inputs and the output. Crisp input values are fuzzified through the proposed membership functions and transformed into values in [0,1]. The output is the social trust value. The linguistic representation of the discussed parameters is *Low*, *Medium* and *High*. The final result is retrieved by the defuzzification process. The degrees of membership of inputs are combined to get the degree of membership of the output variable. For the defuzzification phase, we adopt the *Center of Gravity* (CoG) method [4]. Finally, it should be noted that fuzzy sets and membership functions are defined by experts.

4.3 The Individual Trust Sub-System

In this sub-system, every entity has the ability to use individual experiences to combine them with the knowledge it acquires from the rest entities active in the community. Recall that entities are able to keep locally the historical interactions with other entities as well as the outcome of these interactions. The calculation of the individual trust follows a similar methodology to that of the social trust. The difference lies in the fact that the weight factor concerning the individual experience of the entity making the evaluation is missing. This is natural since in this case the individual history of the entity is used for each of the parameters of the interactions (quality, communication). The individual trust is calculated as

$$\mathrm{IT}_{\mathrm{s_i}} = \frac{\sum_{\mathrm{j=1}}^{\mathrm{n}}(\mathrm{T_j}\cdot \mathrm{F})}{\mathrm{n}} \tag{5}$$

where IT_{s_i} represents individual trust for IA s_i, T_j and F are the weight and the fuzzy value respectively of IA's opinion for s_i for each past interaction, as defined above. In fact, F is the result of the FL individual trust sub-system. Parameter n indicates the number of past transactions with the specific IA. If no past experiences are present, then the individual trust value is set to 0 (a neutral value). As in the social trust sub-system, we use triangular membership functions for inputs and output in the interval [-1,1]. The linguistic values of them are *Low*, *Medium* and *High*. The fuzzification and defuzzification processes are as in the social trust sub-system while membership functions and fuzzy rules are defined by experts.

4.4 The Final Trust Calculation

The IA, after the calculation of social and individual trust, adopts a weighted sum for the final trust value, using a FL system for the extraction of the weight for social trust. Figure 2 shows the architecture of the system. In this rationale, the final trust is calculated as

$$\mathrm{T}_{\mathrm{s_i}} = \mathrm{w_s}\cdot \mathrm{ST}_{\mathrm{s_i}} + (1-\mathrm{w_s})\cdot \mathrm{IT}_{\mathrm{s_i}} \tag{6}$$

where $\mathrm{T}_{\mathrm{s_i}}$ is the final trust for s_i, ST_{s_i}represents the social trust of s_i, $\mathrm{IT}_{\mathrm{s_i}}$ represents the individual trust and w_s is the weight for the social trust calculated by the proposed FL sub-system. The input variables for the third FL sub-system are (i) the total number of social referrals made for the specific entity (SR), (ii) the total number of individual past transactions with the specific entity (IR) and (iii) an error value (er). The error is calculated as

$$\mathrm{er} = \mathrm{ST}_{\mathrm{s_i}} - \mathrm{IT}_{\mathrm{s_i}} \tag{7}$$

and gets values in [-2,2]. The output variable of the FL final trust sub-system is the social trust weight (w_s). In this FL sub-system, we also adopt triangular membership functions for inputs and output and

the linguistic values for them are *Low*, *Medium* and *High*. Table 1 presents the FL rule base for the social weight definition. Membership functions and fuzzy rules are defined by experts while fuzzification and defuzzification are applied as in the previous sub-systems.

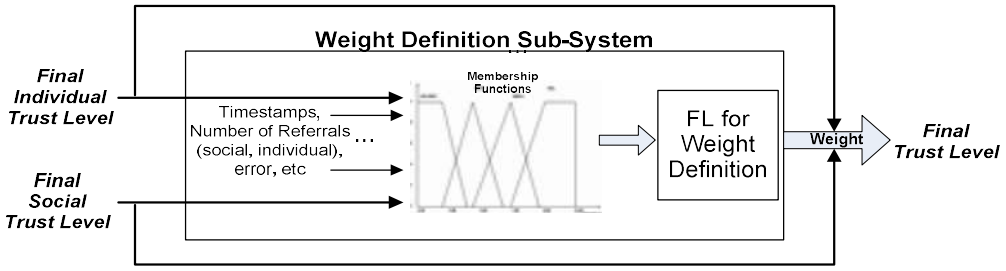

Figure 2. The architecture of the sub-system for the final trust value extraction.

Table 1. FL rule base for final trust calculation.

SR	IR	er	w_s
Low	Any value	Any value	Low
Any value	Low	Any value	High
High	Any value	Low	High
High	Any value	Medium	Medium
Low	Any value	Medium	Low
Any value	High	High	Low
Medium or High	Medium	High	Medium

5 Experimental Evaluation

We evaluate our system with a comparative assessment with another trust calculation model based on [16]. We adopt the *Root Means Square Error* (RMSE), defined as

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{n}} \quad (8)$$

where y_i and $\hat{y}_i$ are the actual and estimated value, respectively. We insert 100 referrals for 20 entities into our system and consider that 50 IAs participate in a pervasive computing application. Referrals for speed and quality of interaction are randomly generated in [-1,1]. We run the system and take the final trust value $\hat{y}_i$ for each of the 20 entities. Then, we consider deception values for every referral. Deception values are updates in true values, negative and positive. Negative deception is realized by adding -0.1, -0.2, -0.3, -0.4, and -0.5 to normal referrals. Positive deception is calculated by adding 0.1, 0.2, 0.3, 0.4, and 0.5 to normal referrals. When we apply a deception value, we retrieve the final trust (y_i) from the system and calculate the error.

Fig. 3 and Fig. 4 show our results for negative and positive deception. In these plots, we use $d_{max} \in \{50, 100\}$ (days) and see how the parameter affects RMSE value. When d_{max}=100 results are similar to [16]. However, when d_{max}=50 our results outperform results in [16]. For example, for deception of -0.1 our system gives RMSE=2.6% while in [16] the result is over 3% for the EFL model and over 4% for the EWL model (approximately). For deception of 0.1 the results are 2.93% and over 3% (EFL model) and -4% (EWL model) respectively. For deception values of -0.5 and 0.5, the results

are 9.71%, 9.83% for our system and close to 15% (EFL model) and -19% (EWL model) for the system presented in [16].

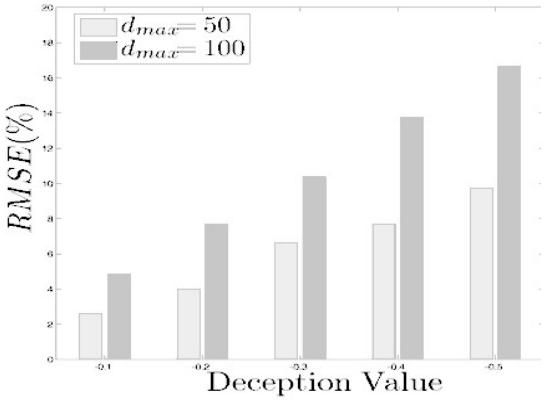

Fig. 3. RMSE for negative deception values.

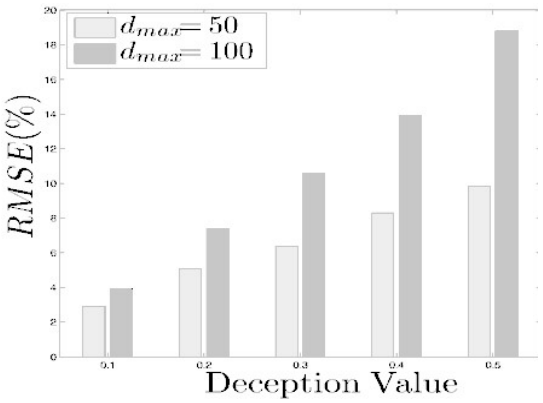

Fig. 4. RMSE for positive deception values.

Finally, Table 2 presents our results for parameter Δ_T. Δ_T shows how d_{max} affects RMSE value. Δ_T is given by $\Delta_T = \frac{T_F - T_S}{T_S} \cdot 100\%$, where TF and TS are the trust value calculated with d_{max} = 50 and d_{max} = 100 days respectively. We see that a high d_{max} lead to a high RMSE. This means that the fluctuation of referrals age negatively affects the trust value and RMSE. When d_{max} is low, the IA takes into account only 'fresh' referrals, thus, it is less affected by possible deceptions.

Table 2. Δ_T results for negative and positive deception values.

Deception Value	Δ_T	Deception Value	Δ_T
-0.5	71.27%	0.1	34.47%
-0.4	78.31%	0.2	45.10%
-0.3	56.95%	0.3	65.57%
-0.2	91.98%	0.4	67.59%
-0.1	86.92%	0.5	91.35%

We perform an additional set of experiments where experts have defined the most trusted 'target entity' for a number of entities participating in the community. In this case, the proposed system should correctly identify the same trusted entity. Experts are a group of people who define the trust value for each examined entity. We consider 4 entities (e_1, e_2, e_3, e_4) that make referrals for 3 'target entities' (s_1, s_2, s_3). These four entities are advised by entity A in order to finalize its behavior concerning the final opinion about the 'target entities'. Table 3 describes the reputation of the entities making referrals for 'target entities' and Table 4 presents the individual experience of A with each of the entities making referrals. In these Tables, values close to zero describe untrusted entities or the absence of experiences while values close to 1 describe trusted entities.

Table 3. Entities' reputation level.

Entity	Reputation Value
e_1	0.40
e_2	0.40
e_3	1.00
e_4	0.70

Table 4. Individual experience of the buyer with the discussed entities.

Entity	Individual experience
e_1	0.00
e_2	0.30
e_3	0.50
e_4	0.90

In Table 5, we can see the social referrals made by these entities for our experimental scenario. For example, referral 1 relates to the referral made by e_1 for s_1 for the parameters described above. The specific referral shows that entity e_1 was satisfied by the quality of interaction with s_1 as well as by their communication (the provided values are close to 1). Moreover, the parameter D is equal to 3, which means that the referral was made 3 days ago. However, from Table 3, we can see that entity e_1 has not a high reputation in the community. Table 6 presents individual experiences with the discussed 'target entities'. From Table 6, we can see that the entity A is satisfied by s_1 compared to the remaining entities (parameters values are close to 1). In our experiments, we consider that d_{max} is equal to 20 days. This means that referrals older than 20 days do not affect the trust calculation process. From Tables 3, 4, 5 and 6, experts agree that the most trusted 'target entity' is s_1. The reason is that s_1 has good referrals from entities participating in the community (from e_3 which has high reputation and A has an experience with it equal to 0.5) and from individual experience (first row of Table 6) compared to the rest.

The described referrals are fed into our system and the calculated final trust value is used to find the most trusted 'target entity'. In our experiments, we focus on the parameter α adopted to conclude R (see above in the section where the social trust calculation is discussed). Parameter α is very important as it affects the weight of the individual experience of A with the entity making a referral. The greater α is the more weight A assigns to the opinion of the entity making the referral. For example, if α =0.6 it means that the individual experience of A with an entity making a referral affects 60% of the final reputation value of the entity (the rest 40% is devoted to the opinion of the society – reputation defined by other members of the community). We examine three cases concerning the value of parameter α:

- Case 1: we take α=0.0 when individual experience is below 0.3, α=0.4 when individual experience is in the interval [0.3, 0.8) and α=0.6 when individual experience is in the interval [0.8, 1].
- Case 2: we take α=0.0 when individual experience is below 0.5 and α=0.6 when individual experience is above or equal to 0.5. This means that A pays no attention in the opinion of entities with which he/she has limited individual experience (value below 0.5).
- Case 3: we take α=0.0 when individual experience is below 0.3, α=0.4 when individual experience is in the interval [0.3, 0.5) and α=0.6 when individual experience is in the interval [0.5, 1].

Table 5. Social referrals for the first experimental scenario[a].

Referral	Entity	Target Entity	Q	C	D (in days)
1	e_1	s_1	0.80	0.80	3
2	e_2	s_1	0.90	0.70	5
3	e_3	s_1	0.40	0.80	10
4	e_4	s_2	- 0.30	0.60	9
5	e_2	s_2	0.40	- 0.70	2
6	e_1	s_3	0.70	0.50	4

Table 6. Individual referrals for the first experimental scenario.

Target Entity	Q	C	D (in days)
s_1	0.90	0.70	1
s_2	-0.80	0.50	6
s_3	0.20	- 0.20	10

Tables 7, 8 and 9 present our results for Cases 1, 2 and 3. As we can see, in all cases the 'target entity' s_1 is more trusted for A than others. In the first case, the s_1 has a final trust value equal to 0.44 while in the other cases the final score is 0.37 and 0.61, respectively. From these results, we can see that the proposed system agrees with the opinion of experts about the most trusted 'target entity'.

Table 7. Results for Case 1.

Target Entity	Social Trust	Individual Trust	w_s	Final Trust
s_1	0.16	0.62	0.40	0.44
s_2	-0.08	0.08	0.42	0.01
s_3	0.50	0.03	0.52	0.27

Table 8. Results for Case 2.

Target Entity	Social Trust	Individual Trust	w_s	Final Trust
s_1	0.12	0.55	0.40	0.37
s_2	-0.04	0.06	0.42	0.01
s_3	0.41	0.02	0.52	0.22

Table 9. Results for Case 3.

Target Entity	Social Trust	Individual Trust	w_s	Final Trust
s_1	0.03	036	0.39	0.61
s_2	0.03	0.01	0.41	0.59
s_3	0.19	0.00	0.50	0.50

In Table 10, we can see results related to the Case 1, however, we utilize the parameter d_{max} to be equal to 10 days. In this case, s_1 has the greatest trust value compared to the rest. It is worth noting that the

[a] Q: quality, C: communication, D: days from the date of the referral

presented trust values are smaller than in cases where d_{max} is equal to 20. Increasing the value of d_{max} means that more referrals are taken into account, thus, they affect more the final result.

Table 10. Results for Case 1 when d_{max}=10.

Target Entity	Social Trust	Individual Trust	w_s	Final Trust
s_1	0.04	0.30	0.39	0.25
s_2	0.02	0.03	0.41	0.02
s_3	0.26	0.00	0.50	0.13

6. Conclusions and Future Work

This paper studies trust estimation for entities in pervasive computing applications. In such cases, IAs need an efficient mechanism for calculation of other IAs trust. We propose a FL system comprising three subsystems. The two subsystems determine the trust level of an entity based on referrals or on IA's individual experiences. The third subsystem results weights for each trust category (social, individual). We present the architecture of each sub-system and provide the mathematical formulations for trust calculation. We compare our system with models in literature and provide results. Experiments show that the system outperforms others for specific values of the parameters. We allege that IAs should pay attention on recent referrals in order to have an efficient mechanism for calculating trust levels of other entities. The model could be used in pervasive computing applications in several domains involving exchanges among entities interacting based on reputation and trust.

Acknowledgements

This research received funding from the European's Union Horizon 2020 research and innovation programme under the grant agreement ARESIBO (Augmented Reality Enriched Situation awareness for Border security) No. 833805.

References

[1] Aref, A., Tran, T., 'A Trust Establishment Model in Multi-Agent Systems', in AAAI Workshop: Incentive and Trust in E-communities, 2015.

[2] Bharadwaj, K., and Al-Shamri, M. Y. H., 'Fuzzy computational models for trust and reputation systems', Electronic Commerce Research and Applications, Vol. 8(1), January 2009, pp. 37-47.

[3] Chhogyal, K., et. al., 'A Value-based Trust Assessment Model for Multi-Agent Systems', in proceedings of the 28th International Conference on Artificial Intelligence, (IJCAI), 2019.

[4] Czabanski, R., et. al., 'Introduction to Fuzzy Systems', Theory and Applications of Ordered Fuzzy Numbers, pp. 23-43, 2017.

[5] Daronnat, S., et. al., 'Human-Agent Collaborations: Trust in Negotiating Control', in proceedings of the Workshop Everyday Automation Experience in conjunction with CHI, 2019.

[6] Devitt, S., 'Trustworthiness of Autonomous Systems', Foundations of Trusted Autonomy, pp. 161-184, 2018.

[7] Esfandiari, B., and Chandrasekharan, S., 'On How Agents Make Friends: Mechanisms for trust Acquisition', In the 4th Workshop on Deception Fraud and Trust in Agent Societies, 2001.

[8] Griffiths, M., 'A Fuzzy Approach to Reasoning With Trust, Distrust and Insufficient Trust', In Proceedings of the Cooperative Information Agents Conference, 2006, pp. 360-374.

[9] Hnativ, A., and Ludwig, S. A., 'Evaluation of Trust in an eCommerce Multi-Agent System Using Fuzzy Reasoning', In Proceedings of the FUZ-IEEE, Korea, 2009, pp. 757-763.

[10] Huynh, D., Jennings, N. R., and Shadbolt, N. R., 'Developing an Integrated Trust and Reputation Model for Open Multi-Agent Systems', in Proceedings of the 7th International Workshop on Trust in Agent Societies, New York, USA, 2004, pp. 65-74.

[11] Javanmardi, S., Shojafar, M., Shariatmadari, S., Ahrabi, S. S., 'FR trust: a fuzzy reputation-based model for trust management in semantic P2P grids', International Journal of Grid and Utility Computing (IJGUC), Vol. 6, No. 1, pp. 57-66, 2015

[12] Josang, A., 'Trust and Reputation Systems', in Aldini A. And Gorrieri R. (eds), 'Foundations of Security Analysis and Design IV', FOSAD 2006/2007 Tutorial Lectures, Springer, 2007.

[13] Josang, A., Ismail, R., 'The beta reputation system', In Proceedings of the 15th Bled Electronic Commerce Conference, Bled, Slovenia, 2002.

[14] Kolomvatsos, K., and Hadjiefthymiades, S., 'How Can We trust Agents in Multi-Agent Environments? Techniques and Challenges', Book Chapter in 'Intelligence Integration in Distributed Knowledge Management', ed. Dr. D. Krol, 2008.

[15] Lesani, M. and Montazeri, N., 'Fuzzy Trust Aggregation and Personalized Trust Inference in Virtual Social Networks', Computational Intelligence, Vol. 25(2), 2009, pp. 51-83.

[16] Ludwig, S. A., Pulimi, V. and Hnativ, A., 'Fuzzy Approach for the Evaluation of Trust and Reputation Systems', in Proceedings of FUZZ-IEEE, Korea, 2009.

[17] Marmol, F. G., Marin-Blazquez, J. G., and Perez, G. M., 'Linguistic Fuzzy Logic Enhancement of a Trust Mechanism for Distributed Networks', in the 10th CIT, 2010, pp. 838-845.

[18] McKnight, D. H., and Chervany, 'The Meanings of Trust', Technical Report, University of Minnessota, Management Information Systems Research Center, 1996, Last Revised April 01, 2000.

[19] Nefti, S., Meziane, F., and Kasiran, K., 'A Fuzzy Trust Model for E-Commerce', In Proceedinsg of the 7th IEEE International Conference on E-Commerce Technology, 2005.

[20] Ramchourn, S. D., Huynh, D., and Jennings, N. R., 'Trust in Multi-Agent Systems', The Knowledge Engineering Review, vol. 19(1), 2004, pp. 1-25.

[21] Rasmusson, L., and Jansson, S., 'Simulated Social Control for Secure Internet Commerce', In C. Meadows, editor, In Proceedings of the 1996 New Security Paradigms Workshop, 1996, pp. 18-26.

[22] Sathiyamoorthy, E., Iyengar, N., and Ramachandran, V., 'Mobile Agent Based Trust Management Framework Using Fuzzy Logic in B2C E-Business Environment', IJCTE, vol. 2(2), 2010, 308-312.

[23] Tajeddine, A., Kayssi, A., Chehab, A. and Artail, H., 'Fuzzy reputation-based trust model', Applied Soft Computing, 11, 2011, pp. 345-355.

[24] Wang, Y., and Vassileva, J., 'Bayesian network-based trust model', In Proceedings of the 6th International Workshop on Trust, Privacy, Deception and Fraud in Agent Systems, 2003.

[25] Wang, Y., Hori, Y., and Sakurai, K., 'On Securing Open Networks Through Trust and Reputation – Architecture, Challenges and Solutions', 1st Joint Workshop on Information Security, 2006.

[26] Zytniewski, M., Klement, M., 'Trust in Software Agent Societies', Online Journal of Applied Knowledge Management, vol. 3(1), 2015.