Volume 2 No.8, AUGUST 2011 ISSN 2079-8407
Journal of Emerging Trends in Computing and Information Sciences

http://www.cisjournal.org

Adaptation, Design and Application of a Computational Trust and Reputation Model for the Assessment of an Agent

[1]ETENG, I.E , [2]OSOFISAN, N. O.
[1]Department of Mathematics/Statistics & Computer Science University of Calabar, Calabar, Nigeria
[2]Department of Computer Science University of Ibadan, Ibadan, Nigeria
[1]idongesitessien@yahoo.com, [2]mamoshof@yahoo.co.uk

ABSTRACT

Computational Trust and Reputation models are the framework for measuring trust and reputation in online services, software and general service provision in the field of computing. This paper presents the adaptation, design and application of a Trust and Reputation model called ReGret. The advantage of the ReGreT system over other models is that it takes advantage, among other things, of social relations between agents to overcome the problem that arises from non-availability of information from only previous past direct interactions of the agent. The modular nature of this model is exploited and the model is applied in the practical scenario of assessing an agent. The system design utilizes UML diagrams, algorithms and interfaces to show the ReGret model's activities using a hotel scenario. Implementation is done using an object-oriented programming language; the Java programming language. Recommendations are also given to aid further study in computational trust and reputation models.

Keywords— *Computational Trust, Reputation, Social Networks, Witness Information*

1. INTRODUCTION

The study of trust and reputation has many applications in information and communication technologies. Trust and reputation systems have been recognised as key factors for successful electronic commerce adoption. These systems are used by intelligent software agents both as a mechanism to search for trustworthy exchange partners and as an incentive to decision-making about whether or not to honour contracts. Reputation is used in electronic markets as a trust-enforcing, deterrent and incentive mechanism to avoid cheats and frauds, [1],[2] & [3]. Another important area of application in agent technology is teamwork and cooperation [4].

There are not too many works that give a general view of trust and reputation from the point of view of Computer Science. [3] presents an overview of online reputation mechanisms that are currently used in commercial websites. In the area of trust, [5] examine the various definitions of trust for Internet applications. There are also proposals to establish a topology for reputation and trust, [6] & [7].

Advantages of Trust and Reputation Mechanisms include:

- Each agent is a norm enforcer and is also under surveillance by the others. No central authority is needed.
- The nature of trust and reputation agents allows arriving where laws and central authorities cannot.
- Untrustworthy agents are shut out from further interactions.
- These systems are used by intelligent software agents both as mechanisms to search for trustworthy exchange partners and as incentives in decision-making about whether or not to honour contracts.
- Reputation is used in electronic markets as a trust-enforcing, deterrent, and incentive mechanism to avoid cheats of fraud.

Problems of Trust and Reputation Mechanisms include:

- Exclusion must be the punishment for the outsider.
- Not all kinds of environments are suitable to apply these mechanisms.
- Trust and reputation can be arranged from different perspectives and can be used in a wide range of situations thereby making the classification of trust and reputation models a difficult task.
- Bootstrapping problem. It is a little difficult to establish trust for fresh member agents of a community. Recent research work, however are addressing this.

1.1 Aim of Research

The main aim of this research work is to describe, build upon, adapt and apply a Computational Trust and Reputation Control Mechanism; ReGret [7] in a real life scenario – a Hotel Management Agent. Several Computational Trust and Reputation models exist and they include among others; OpenPGP, Marsh, eBay/OnSale, Sporas and Histos, TrustNet, Fuzzy models, LIAR etc.

This research work will attempt to review some of these models, particular attention; however will be paid to the workability and application of ReGret on a Computer system to illustrate its workability, efficiency, drawbacks, constraints and applicability in an existing system.

1.2 Objectives of Research

The objectives of the research include the following:

- To establish and describe what Computational Trust and Reputation Models are and explain how they are used.
- To give a brief review of Computational Trust, Reputation and their models.
- To demonstrate the workability of Trust and Reputation models.
- To simulate a Computational Trust and reputation Model; Regret; using the Java programming language.
- To encourage further research in the area of Trust and Reputation.

1.3 Definition of terms

- **Trust**: An assumed reliance on some person or thing. A confident dependence on the character, ability, strength or truth of someone, something or an entity.
- **Reputation:** The estimation of the consistency over time of an attribute or entity and the opinion of others over an entity.
- **Computational Trust:** Trust as an evaluative belief.
- **Computational Reputation:** Reputation that adds a collective dimension to the truster.
- **Trust Model:** a collection of rules that inform applications on how to decide the legitimacy of an entity.
- **Trust Management:** Trust management is a unified approach to specifying and interpreting security policies, credentials, relationships which allow direct authorization of security-critical missions.
- **Trust metric:** a measurement of the degree to which one social actor (an individual or a group) trusts another social actor. Trust metrics may be abstracted in a manner that can be implemented on computers.

1.4 Research methodology

The ReGret model studied in this work is numerical in nature and uses direct information, witness information, sociological information, prejudice, and is context-dependent. It is also subjective. This informs the need to utilize research methods that collect, analyse and compute numerical input to yield the desired output. Defined parameters and variables have numerical basis and serve to describe the workings of the ReGret model.

2. LITERATURE REVIEW

2.1 Survey of trust in Internet applications

The migration from centralized information to Internet-based applications means that transactions have to span a range of domains and organizations not all of which may be trusted to the same extent. Inconsistencies in current trust relationships highlight the need for a flexible, general-purpose trust management system that can navigate these domains [5]. Trust specification and management can be used as the starting point for subsequent refinement into security.

2.1.1 Defining trust

Trust is a complex subject relating to belief in honesty, trustworthiness, competence, reliability etc of a trusted person or service. There is no consensus in the literature on what constitutes trust management. [9] in their considerations on the theoretical framework of trust, examine it from the perspective of personality theorists, sociologists, economists and social psychologists. They state that trust as defined in the Webster's dictionary is:

- An assumed reliance on some person or thing. A confident dependence on the character, ability, strength or truth of someone or something.
- A charge or duty imposed in faith or confidence or as a condition of a relationship.
- To place confidence (in an entity).

They highlight the implications of these definitions and combine their results with the psychological perspective of trust to create their definition of trust in a system-"a belief that is influenced by the individual's opinion about certain critical system features".

The European Commission Joint Research Centre defines trust as "the property of a business relationship, such as reliance can be placed on the business partners and the business transactions developed with them". This view of trust is from a business management perspective and offers an interesting analysis of what must be done to enable trust in E-Commerce.

The Oxford Reference Dictionary states that trust is "the firm belief in the reliability or truth or strength of an entity". A trustworthy entity will typically have a high reliability and so will not fail during the course of an interaction, will perform a service or action within a reasonable period of time, will tell the truth and be honest with respect to interactions, and will not disclose confidential information.

Thus, trust is really a composition of many different attributes: reliability, dependability, honesty, trustfulness, security, competence, and timeliness, which may have to be considered depending on the environment

in which trust is being specified. Trust is a vast topic that incorporates trust establishment, trust management and security concerns. The lack of consensus with regards to trust has led authors to use the term trust, authorization and authentication interchangeably. The outcome of a trust decision is based on many things such as the trustor's propensity to trust, its beliefs and past experiences relating to the trustee.

Authorization can be seen as the outcome of refinements of a more abstract trust relationship. For example, if a lecturer develops a trust relationship with a particular student, he/she may authorize the student to install software on his computer and hence set up the necessary access control rights for a subject to perform specific actions on a specific target with well defined constraints.

Trust is then defined as "the firm belief in the competence of an entity to act dependably, securely and reliably within a specified context" (assuming dependability covers reliability and timeliness).

2.1.2 Properties of trust relationships

In general, a **trust relationship is not absolute**- A will never trust B to do any possible action it may choose. A trustor trusts a trustee with respect to its ability to perform a specific action or provide a specific service within a context. Even trust in oneself is not usually absolute and there is need to protect resources one owns from mistakes or accidents he/she may cause. Examples include protecting files from accidental deletion or mechanisms to prevent a person driving a car when under the influence of alcohol.

A trust relationship can be **one-to-one** between two entities; however **it may not be symmetric. A's trust in B is not usually the same as B's trust in A**. It may be a **one-to-many** relationship in that it can apply to a group of entities such as the set of students in a particular year. It can also be **many-to-many** such as mutual trust between members of a group or a committee, or many-to-one such as several departments trusting a corporate head branch. In general, the entities involved in a trust relationship will be distributed and may have no direct knowledge of each other so there is need for mechanisms to support establishment of trust relationship between distributed entities.

There is often a level of trust associated with a relationship; some entities may be trusted more than others with respect to performing an action. If discrete values are used, then a quantitative label such as high, medium or low may be sufficient. Some systems support arithmetic operations on trust recommendations, so numeric quantification is more appropriate. It is also possible to provide a mapping from qualitative to numeric labels.

2.2 Trust management

Blaze et al., [10] define trust management as a unified approach to specifying and interpreting security policies, credentials, relationships which allow direct authorization of security-critical actions'. Implemented automated trust management systems of note include Policy-Maker, KeyNote & REFEREE.

2.2.1 Trust management solutions

Most trust management systems focus on protocols for establishing trust in a particular context. Some make use of a trust policy language to allow the trustor to specify the criteria for a trustee to be considered trustworthy.

2.2.1.1 Policy-based trust

Using policies to establish trust, focuses on managing and exchanging credentials and enforcing access policies. Work in policy-based trust generally assumes that trust is established simply by obtaining a sufficient amount of credentials pertaining to a specific party, and applying the policies to grant that party certain access rights. The recursive problem of trusting the credentials is frequently solved by using a trusted third party to serve as an authority for issuing and verifying credentials. In policy based model a certified third party like **verisign** will issue the digital certificate and the trustworthiness of the client will be identified by checking the proof of identity [11].

A digital certificate is issued by a certification authority and verifies that a public key is owned by a particular entity. The certification authority does not vouch for the key owner, but simply authenticates the owner's identity. This is necessary to establish a resource access or service provision trust relationship and may implicitly reduce the trustor's risk in dealing with the trustee. However, the policy governing what resources or services the trustee is permitted to access is not handled by the certificate infrastructure, but is left up to the application. Some systems that have used public key certificates include the under listed.

- **PGP and X.509**

Two of the main certificate systems dealing with authentication are PGP [12] and X.509 [13]. According to [14], PGP was mainly envisaged for secure e-mail communication. In fact, even now, for this purpose, it has generated a good amount of trust. Of course, it has drawbacks but the trust is for a class of applications or transactions. PGP is a zero-configuration model, i.e. without any pre-configured infrastructural setup, users can participate in it. Users of PGP generate an asymmetric key pair on their own and distribute their public-key through disks or newspapers (or any other off-line medium), so that others can communicate securely with them. Users maintain public keys of other users in a local database, called keyring. The PGP model is used for authentication relating to electronic mail type of applications between human users. It supports a Web of Trust Model in that there is no centralized or hierarchical relationship between

certification authorities as with X.509. The underlying assumptions of the model are that a trustor may trust other entities, may validate certificates from other entities or may trust third parties to validate certificates.

The X.509 trust model is a strictly hierarchical trust model for authentication. Each entity must have a certificate that is signed by the central certification authority or another authority, which has been directly or indirectly certified by it. This model assumes that certification authorities are organised into a universal certification authority tree and that all certificates within a local community will be signed by a certification authority that can be linked into this tree.

Neither of these models can be used to model trust in all domains. Due to PGP's lack of official mechanisms for the creation, acquisition and distribution of certificates, it is considered unreliable for E-Commerce but appropriate for personal communication. X.509's rigid hierarchical structure may lead to unnatural business alliances between competing companies that violate the natural order of trust. Some applications such as the reference information distribution systems need certificates to have a lifespan longer than is currently allowed by either scheme.

- **AT&T PolicyMaker and KeyNote**

PolicyMaker [10] & [15] is a trust management application, developed at AT&T Research Laboratories, that specifies what a public key is authorised to do. Traditional certificate frameworks such as PGP and X.509 do not bind access rights to the owner of the public within the certificate framework. Schemes such as these require a two-step process:

a) The binding of a public key to its owner, which occurs within the certificate framework, and
b) The binding of access rights to the identified key owner, which occurs outside the certificate framework.

The PolicyMaker system is essentially a query engine which can either be built into applications (through a linked library) or run as a 'daemon' service. The inputs to the PolicyMaker interpreter are the **local policy**, the **received credentials** and an **action string** (which specifies the actions that the public key wants to perform). The interpreter's response to the application can either be yes or no or a list of restrictions that would make the action acceptable. A policy is a trust assertion that is made by the local system and is unconditionally trusted by the system. A credential is a signed trust assertion made by other entities and signatures must be verified before the credentials can be used.

KeyNote [15], the successor to PolicyMaker, was developed to improve on the weaknesses of PolicyMaker by AT&T Research laboratories. It has the same design principles of assertions and queries but includes two additional design goals, namely: standardization and ease of integration. In KeyNote, more is done in the trust management engine, rather than in the calling application (as was the case in PolicyMaker). Signature verification is done in the KeyNote engine and a specific assertion language is used-this allows simpler integration with its compliance seeker. The KeyNote engine is passed a list of credentials, policies, the public keys of the requester and an 'Action Environment' (which is essentially a list of attribute value pairs) by the calling application.

2.2.1.2 Reputation-based trust

Using reputation to establish trust, where past interactions or performance for an entity are combined to access its future behaviour. Research in reputation trust uses the history of an entity's actions/behaviour to compute trust, and may use referral-based trust (information from others) in the absence of (or in addition to) first-hand knowledge. In the latter case, work is being done to compute trust over social networks (a graph where vertices are people and edges denote a social relationship between people), or across paths of trust (where two parties may not have direct trust information about each other, and must rely on a third party). Recommendations are trust decisions made by other users, and combining these decisions to synthesize a new one, often personalized, is another commonly addressed problem.

Recently, formal models have been proposed for reputation-based trust management. In contrast to credential-based trust management, an agent's reputation serves as the basis for trust [16].

2.2.1.3 General models of trust

There is a wealth of modelling and defining trust, its prerequisites, conditions, components and consequences. Trust models are useful for analyzing human and agenized trust decisions and for operating computable models of trust. Work in modelling trust describes values or factors that play a role in comprising trust and leans more on work in psychology and sociology for a decomposition of what trust comprises. Modelling research ranges from simples access control policies (which specify who to trust to access data or resources) to analyses of competence, beliefs, risk, importance, utility, etc. these subcomponents underlying trust in understanding the more subtle and complex aspects of composing, capturing, and using trust in a computational setting.

Many have recognized the value of modelling and reasoning about trust computationally. [17], [18],[19],[20]. A wide variety of literature now exists on trust, ranging from specific applications to general models. However, the meaning of trust as used by each researcher differs across the span of existing work. Three definitions of trust from existing research are given as reference point for understanding trust.

2.2.1.4 Trust in information resources

Trust is an increasingly common theme in web related research regarding whether web resources and websites are reliable. Moreover, trust on the web has its range of varying uses and meanings, including capturing ratings from users about the quality of information and services they have used, how web site design influence trust on content and content providers, propagating trust over links, etc.

2.3 The Regret model

The ReGreT model is a modular trust and reputation model oriented to complex e-commerce environments where social relations play an important role.

The advantage of the Regret System over other existing trust and computational models is its ability to integrate a third source of information – **information from social networks.** Prior to the development of the Regret system, most models of trust and reputation have considered two basic information sources: i) the direct interaction among agents and ii) information provided by other agents about their past experiences.

According to [8], the major characteristics of the ReGreT model includes the following:

- It takes into account direct experiences, information from third party agents and social structures to calculate trust, reputation and credibility values.
- It has a trust model based on direct experiences and reputation.
- It incorporates an advanced reputation model that works with transmitted and social knowledge.
- It has a credibility module to evaluate the truthfulness of information received from third party agents.
- It uses social network analysis to improve the knowledge about the surrounding society.
- It provides a degree of reliability for the trust, reputation and credibility values.
- It can adapt to situations of partial information and improve gradually its accuracy when new information becomes available.
- It can manage at the same time different trust and reputation values associated to different behavioural aspects.

The architecture of the ReGreT system is such that among other components, it maintains three knowledge bases:

i) The Outcomes Database (ODB) where the agent stores previous contracts and their results.

ii) The Information Database (IDB) that is used as a container for the information received.

iii) Sociograms Database (SD) where the agents stores the sociograms that defines its social view of the world.

These databases feed the different modules of the system.

The **trust model** therefore consists of the **Direct Trust module** and the **Reputation model**.

The reputation model on the other hand can use three distinct and disjoint approaches: **witness reputation, neighbourhood reputation** and **system reputation**. Details of the approaches are documented in Sabater (2009). Note, however, that the modular nature of the system allows the system to choose either of the three reputation approaches. It also allows for adaptability to different levels of knowledge.

In the ReGreT system, each trust and reputation value has an associated reliability measure. This measure tells the agent how confident the system is on the value according to how it has been calculated.

The ReGret Trust and Reputation model uses a set of mathematical models to determine the reputation of a selected agent based on impression ratings that have been recorded for that agent. The proposed system focuses on the computation of the subjective reputation for a given agent (in this case, the agent is a hotel establishment).

The model for the subjective reputation is defined as follows:

$$T_{i,j(t)} = \sum_{Wk \in R_{i,j}} \phi(t,t_k).W_k \qquad eqn\ (1)$$

Where $T_{i,j}$ (t) is the computed reputation value at time t,
t is the current time,
W_k is the rating obtained from the impression K recorded about the agent under evaluation
t_k is the time at which W_k was recorded,
$R_{i,j}$ is the set of all W_k obtained from the set of all impressions about the agent.
Each computed value in the right hand side of the above model is considered a rating factor.
The function ϕ (t, t_k) is defined as

$$\phi\ (t,tk) = \frac{f(t_k,t)}{\sum WLE \in R_{i,j} f(t_k,t)} \qquad eqn\ (2)$$

And f (t_k, t) = $\frac{t_k}{t}$

f (t_k,t) is called the rating recency function

The rating recency function ensures that impression ratings that have been recorded recently have more relevance than impressions recorded at an earlier time.

The general outline for the subjective reputation model is as follows:

- Obtain the set of impressions about the given agent.
- For every impression in the impression set compute a rating factor based on the time the impression was recorded and the rating ssigned to that impression.
- Sum all computed rating factors to obtain the reputation value

Besides the reputation value, it is important to know how reliable that value is. Although, there are a number of elements that can be taken into account when calculating how reliable a subjective reputation value is, the proposed system will focus on the number of impressions used to calculate the reputation value.
The following model is used to obtain the degree of reliability.

$$No(R_{ij}) = \begin{pmatrix} \sin(\frac{\pi . R_{ij}}{2.itm}) \\ 1 \end{pmatrix} otherwise \quad | R_{ij} | \leq \text{itm}$$

Where R_{ij} is the number of impressions used in calculating the subjective reputation,

Where itm is the maximum number of impressions; over which the reputation value is considered to be very reliable i.e. if the number of impressions used in computing the reputation value is greater than the value of itm, then the degree of reliability of the computed value is 100%.

It is important to note that the value of itm is application dependent. For the proposed system, itm is set at 7. Thus, any reputation value computed using more than 7 impressions is considered very reliable.

2.2.1 Impressions

The Regret model defines an impression as the subjective evaluation made in a certain aspect of an outcome. An impression is represented by a tuple of the form.

I = (a, b, o, ø, t, w).

Where **a,** is the agent making the judgement in the context of the proposed system, this is the application user;

b is the agent being judged in the context of the proposed system which is the hotel organization/establishment;
o is the outcome of the interaction between the two agents ;
ø represents the aspect of the outcome being judged. For the proposed system, this could be either the Hotel's Quality or the distance from the hotel's location to the user's business destination.
t represents the time at which the impression is being recorded.
w is the rating assigned to the impression by the user. For the proposed system, the values range from terrible (-1) to excellent (1).

3. DESIGN AND METHODOLOGY

The design of a proposed system based on the ReGret trust and reputation model is illustrated here. Class diagrams, a use case diagram, interface design, etc are used to highlight the ReGret model and describe it in detail. The system being designed is applied in a hotel scenario.

3.1 System requirement specification

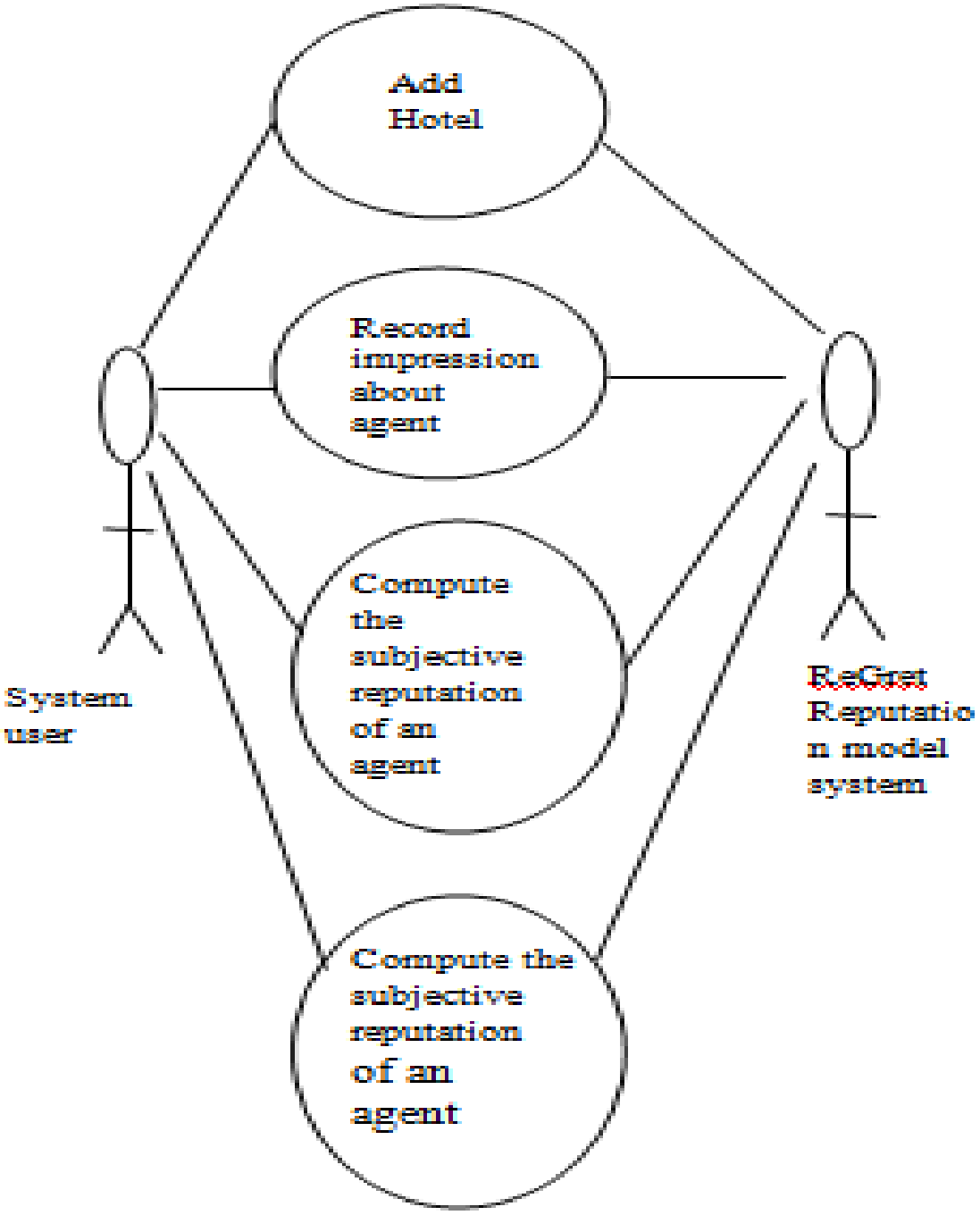
The system requirements specification defines the set of functions or tasks that a given system must be capable of performing. These specifications serve as a blue print for the proposed system and provide the developer(s) with information regarding the features that must be included in the system for it (the system) to be considered as serving its purpose.

The following are the requirements for the proposed system.

- The system should allow users to add new agents (in the application's context the agents are Hotel organizations) to the system's database.
- Users should be able to input their impressions about selected agents in the database.
- Users should be able to assess the reputation of a selected agent based on their impression about that agent as available in the system's database.
- Users should also be able to specify the maximum number of impressions required for a reputation assessment to be considered 100% accurate.
- The system should provide a default value for the number of impression used in the computation of the reliability of a reputation assessment in the event where the user fails to specify one, as mentioned above.

3.2 Use case diagram

The use case diagram is a UML tool that provides a graphical representation of the functional system requirements of a given system. It provides an overview of the system based on the functions or tasks a user can perform when interacting with the system. The use case diagram for the proposed system is given below.



Use case diagram

3.3 UML class diagram

The class diagram provides an overview of the target system by describing the objects and classes in the system and the relationships between them. The class diagram for the proposed system is given below.

Volume 2 No.8, AUGUST 2011 ISSN 2079-8407
Journal of Emerging Trends in Computing and Information Sciences

http://www.cisjournal.org

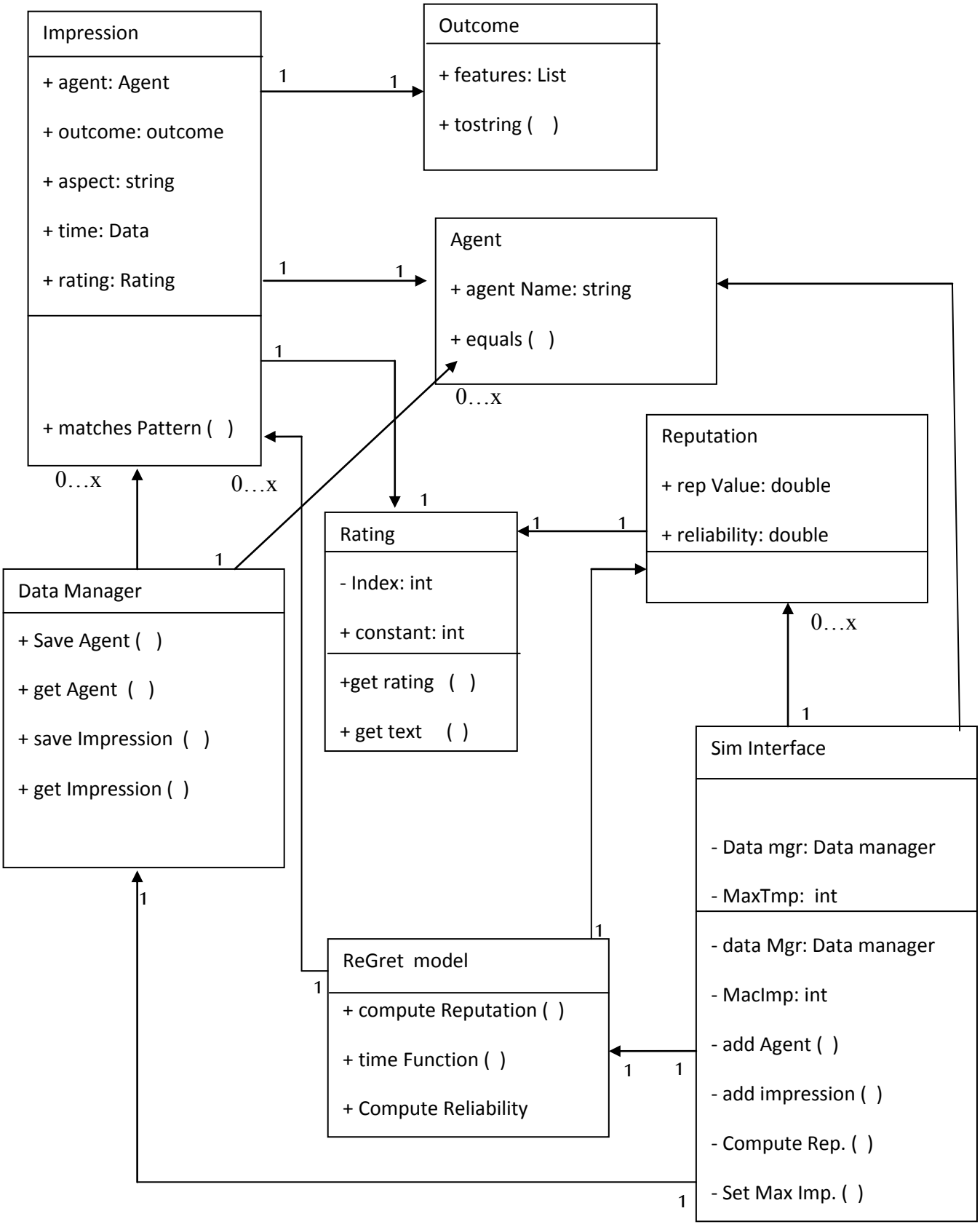


Fig 3(b): UML Class diagram

Data manager class

This class is responsible for the file input/output operations of the system. It handles the storage and retrieval of the various information used by the application.

- **Outcome class**

The outcome class serves as a data structure for storing the outcome of the interaction between a user and a given hotel.

- **Agent class**

This serves as a data structure for storing information about a give agent (in this case, a hotel). For this system, the information stored is the hotel's name.

- **Rating class**

The rating class represent the ratings for a given user impression. It encapsulates the range of possible rating values in addition to the value selected by the user.

- **Impression class**

The impression class represents a user's impression about a hotel. It encapsulates values such as the target hotel, the outcome of the interaction between the user and the hotel, the data the impression was recorded and the rating for that particular impression.

- **Reputation class**

This class holds information about the reputation that is competed for a given hotel. The information includes the value of the reputation, in addition to the reliability value for the computed reputation.

- **Regret model class**

This class models the regret model. It provides utility functions that enable the system to compute the subjective reputation using the ReGret Trust and reputation model.

- **SimInterface**
-

This class contains the processing logic of the system. It also encapsulates the graphical user interface for the system and handles the interaction between the various classes that make up the system.

3.4 Interface design

The proposed system will make use of a graphical user interface (GUI). The prototype design for the various interface screens are given below

Fig 3(c): Application screen design

This interface is the main screen of the application. A user gains access to the system's functions through the command buttons available on this interface. The command buttons and their functions are described below.

✓ **Add Hotel:** This allows a user to add the name of a new hotel to the application database.

✓ **New Impression**: this allows a user to record a new impression about a given hotel on the user's interaction with the hotel.

✓ **Compute reputation**: This allows the user to compute the subjective reputation for a given hotel.

✓ **About:** This provides information about the purpose of the application.

✓ **Exit:** This terminates the application.

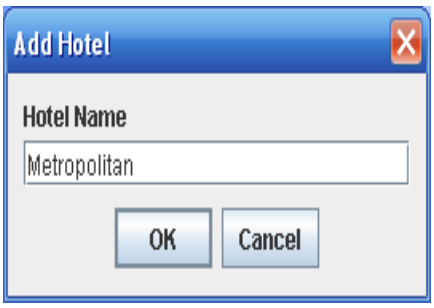


Fig 3(d): Add Hotel Input screen

This is the interface where the user adds a new hotel to the application's database. It provides a text box where the user can input the name of the hotel.

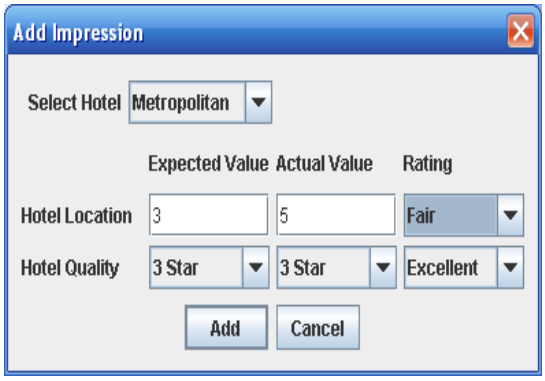


Fig 3(e): Add Impression Input screen

The user adds new impressions using this interface. When creating the impression, the user specifies the following information:

- The hotel under consideration, selected from the 'Select Hotel' drop-down list.
- For each hotel feature (i.e. Hotel location and Quality), the user specifies the expected value for that feature, the actual value for the feature and the rating for the feature based on the user's impression about the interaction.

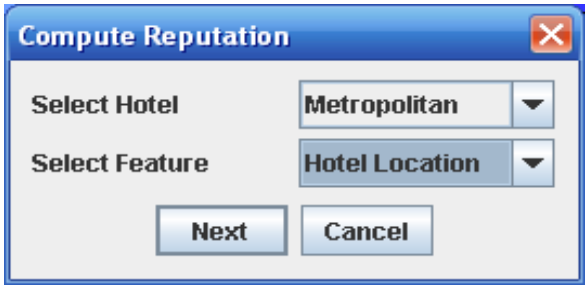


Fig 3(f): Compute Reputation input screen

This is the interface through which the user specifies the parameters to be used in computing the subjective reputation for a given hotel. The interface provides a drop-down list where the user selects the hotel feature on which the analysis will be based.

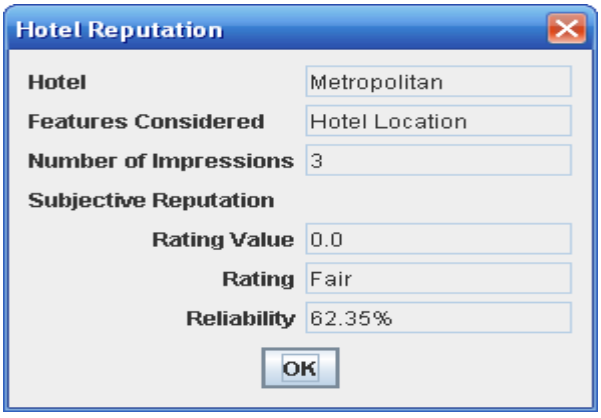


Fig 3(g): Compute Reputation Output screen

This interface displays the result of a subjective reputation computation. The output includes the following information:

- ✓ The name of the hotel;
- ✓ The features considered during the computation;
- ✓ The number of impressions used in the computation;
- ✓ The computed rating for the subjective reputation;
- ✓ The reliability of the rating based on the number of impression used.

The ReGret algorithm:

i. Set sum = 0
ii. Obtain the current time
iii. Select an impression record
iv. Obtain the recorded time for the selected record.
v. Computer the rating factor ø (t, t_k) using the current time and recorded time as parameters
vi. Multiply the rating factor by the recorded rating for he selected record.
vii. Add the obtained result to sum.
viii. If impression records are still available then repeat steps 3 to 7.
ix. Subjective reputation value = value of sum.

4. SUMMARY & RECOMMENDATIONS

4.1 Summary and recommendations

The aim of this paper has been the simulation of a computational trust and reputation model-Regret. The preceding sections have highlighted various aspects of the simulation process. This work is by no means the end of simulation systems based on the ReGret model but serves as exposition to other such systems. This seeks to serve as stimulus to further research in Trust and Reputation.

Below are a number of recommendations for further research. They include:

- Good mechanisms to increase the efficiency of actual trust and reputation model through the introduction of sociological aspects to trust and reputation models.
- Regret system proposes methods to combine different sources of information. These are far from being a general solution. A solution for this could be the use of static methods i.e. adaptive methods that can modify how to combine different sources of information according to the environment. This is not an easy task and requires further study.
- Consensus is being reached on what trust is and what reputation is respectively. There are several works that help to give a precise and

distinct meaning to both concepts; the relation between both concepts needs to be studied in detail.

- There are a very little number of test-beds and frameworks to evaluate and compare model under a set of representation and common conditions. This situation is quite confusing especially for possible users of these models.

It is thus urgent to define a set of test-beds that allow the research community to establish comparisons in a similar way to what obtains in other areas e.g. machine learning etc.

REFERENCES

[1] eBay. http://www.eBay.com.

[2] Amazon: 2002, Amazon Auctions. http://auctions.amazon.com

[3] Dellarocas, C.: 2003, 'The digitalization of Word-Of-Mouth: Promise and Challenges of Online Reputation Mechanisms'. Management Science.

[4] Montaner, M., Lopez, B. and dela Rosa, J.: 2002, Developing trust in recommender agents. In: Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS-02), Bologna, Italy. pp. 304-305

[5] Grandison T. and Sloman, M.: 2000, A Survey of Trust in Internet Applications. Department of Computing, Imperial College, London.

[6] Mui, L., Mohtashemi, M. and Halberstadt, A.: 2002, Notions of Reputation in Multi-Agent Systems: A Review.

[7] McKnight, D. H. and Chervany, N. L.: 2002, Notions of Reputation in Multi-Agent Systems: A Review.

[8] Sabater, J. and C. Sierra: 2002, 'Reputation and Social Network Analysis in Multi-Agent Systems'. In: Proceedings of the first international joint conference on autonomous agents and multiagent systems (AAMAS-02), Bologna, Italy. pp. 475—482.9.

[9] Kini, A. and J. Choobineh (1998) "Trust in Electronic Commerce: Definition and Theoretical Considerations", 31st Annual Hawaii Int'l. Conf. System Sciences, 1998, Hawaii.

[10] Blaze M., Feigenbaum J. and Lacy J.: 1996, Decentralized Trust Management, In: IEEE Conference on Security and Privacy, Oakland, California,

[11] Sathiyamoorthy 1, N.Ch.Sriman Narayana Iyenger & Ramachandran (2005).Trust management for e-transactions Vol. 30, Parts 2 & 3 pp. 141–158.

[12] Zimmermann P. (1995) PGP source code and internals (Cambridge, MA: MIT Press)

[13] FordW, BaumMS 2002 Secure electronic commerce: Building the infrastructure for digital signatures and encryption 2nd edn (Englewood Cliffs, NJ: Prentice Hall)

[14] Vishwas Patil & Shyamasundar R. K. School of Technology and Computer Science, Tata Institute of Fundamental Research, Homi Bhabha Road, Colaba, Mumbai.

[15] Blaze M, Feigenbaum J, Ioannidis J, Keromytis A 1999 The KeyNote Trust-management system. Version 2. RFC 2704, Internet Engineering Task Force

[16] Shmatikov, V., Talcott, C.L.: Reputation-based trust management. Journal of Computer Security 13(1), 167–190 (2005)

[17] Longo L. (2007). Security Through Collaboration in Global Computing: a Computational Trust Model Based on Temporal Factors to Evaluate Trustworthiness of Virtual Identities. Master Degree, Insubria University.

[18] D. Quercia, S. Hailes, L. Capra (2006) (PDF). B-trust: Bayesian Trust Framework for Pervasive Computing. iTrust. http://www.cs.ucl.ac.uk/staff/d.quercia/publications/querciaB-trust06.pdf.

[19] Seigneur J.M., (2006). Seigneur J.M., Ambitrust? Immutable and Context Aware Trust Fusion.. Technical Report, Univ. of Geneva,.

[20] Staab, E. and Engel, T. (2009). Tuning Evidence-Based Trust Models. In: Proceedings of the 2009 IEEE International Conference on Information Privacy, Security, Risk and Trust (PASSAT'09), Vancouver, Canada, pp. 92-99, IEEE. pp. 92–99. doi:10.1109/CSE.2009.209.