DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

A FORMAL TRUST MODEL BASED ON RECOMMENDATIONS

by

Mahir KUTAY

January, 2013
İZMİR

A FORMAL TRUST MODEL BASED ON RECOMMENDATIONS

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor of
Philosophy in Electrical-Electronic Engineering, Electrical-Electronics Program

by
Mahir KUTAY

January, 2013
İZMİR

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"A FORMAL TRUST MODEL BASED ON RECOMMENDATIONS"** completed by **MAHİR KUTAY** under supervision of **YRD. DOÇ. DR. ZAFER DİCLE** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.

Yrd. Doç. Dr. Zafer DİCLE

Supervisor

Prof. Dr. Mehmet Ufuk ÇAĞLAYAN

Thesis Committee Member

Prof.Dr. Mustafa GÜNDÜZALP

Thesis Committee Member

Yrd. Doç. Dr. Yavuz ŞENOL

Examining Committee Member

Prof. Dr. Emin ANARIM

Examining Committee Member

Prof.Dr. Mustafa SABUNCU

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

Dedicated to the memory of my beloved father Mahmut Bedii Kutay.

I am deeply grateful to my co-advisor Professor Mehmet Ufuk Çağlayan for his invaluable motivating guidance and support during the entire course of this study.

I also wish to express my deep and sincere gratitude to my advisor Assistant Professor Zafer Dicle.

I would like to thank to Professor Mustafa Gündüzalp for his motivating guidance.

I am very grateful to Dr. Şerif Bahtiyar for his support during the paper preparation and correction courses.

I would like to thank Mrs. Ayşe Nuriye Yılmaz and Mrs. Sibel Bezci Ercan for their support during the coding process of TAST software tool.

And finally I am immensely grateful to my beloved wife Belkıs and daughter Özge for their endless understanding and support.

Mahir Kutay

A FORMAL TRUST MODEL BASED ON RECOMMENDATIONS

ABSTRACT

A modern society is based on the division of labor and people inevitably rely on others. Improvement in technology makes it possible to perform economical transactions between partners living in different geographical locations and who may never see each other during their life-span. Recommender systems guide people to buy goods materials based on information from other people. A large set of alternative ways to organize such systems exists. The information that other people provide may come from explicitly from ratings, tags, reviews, or implicitly from how they spend their time and money. The information obtained can be used to select, filter, or sort items.

This thesis examines formal trust assessment models. Main contributions of the thesis can be summarized as following:

- A formal model to assess the trust to the organizations in a specified context-set by using web-based survey data is developed. Addition of importance parameter to trust calculations and calculation of trust in real-number intervals by selected confidence probability are the main contributions.
- Trust and confidence propogation in trust chains are investigated. Propogation of confidence is here the main contribution.
- Trust and confidence propogation in service oriented systems are modeled. Propogation of confidence in service-oriented systems is again the main contribution in this model.
- A software tool called Trust Assesment Software Tool (TAST) has been developed. This is a flexible program that can be applied to the organizations working in the same business-field. TAST calculates the trust assessments of the organizations in selected time intervals. TAST can make trust assessment comparisons by competitor organizations in selected time intervals.
- We also show the applicability of our contributions by examples and case studies.

Keywords : trust, web, bi-partite graph, recommendation, trust metric, trust management, trust transitivity, web-based survey, trust chain.

ÖNERİLERE DAYALI FORMAL BİR GÜVEN MODELİ

ÖZ

Günümüz toplumu iş bölümüne dayanmaktadır, bunun kaçınılmaz bir sonucu olarak insanlar birbirlerine bağlı olarak çalışmak zorundadırlar. Teknolojik gelişmeler, değişik coğrafi bölgelerde yaşayan ve birlerini ömürleri süresince belki de hiç göremeyecek olan insanların ticaret yapmasını mümkün kılmıştır. Öneri sistemleri insanların deneyimlerini diğerlerine aktarmalarını ve tercihlerini yönlendirmelerini sağlarlar. Öneri sistemleri tercihlerin belirlemesinde önemli bir rol oynar. Öneri sistemleri oluşturmanın oldukça geniş seçenekleri vardır. Bilgi diğer insanlardan anketler, oylamalar, eleştiriler gibi doğrudan yollarla ya da insanların zaman ve para harcama yöntemlerinin izlenmesiyle dolaylı yollardan elde edilir. Elde edilen bilgi tercihlerin önem sırasına göre sınıflandırılmasını ve yönlendirilmesini sağlar.

Bu tez biçimsel güven hesaplama modellerini incelemektedir. Tezin başlıca katkıları aşağıda özetlenmiştir:

- Web üzerinden yapılan anketler yoluyla toplanan bilgiyi kullanarak, tanımlanan bir içerik kümesi için kuruluşlara olan güveni hesaplayan biçimsel güven modeli geliştirilmiştir. Güven hesaplamalarına önem değişkeninin eklenmesi ve güvenin seçilen güven olasılığına göre gerçel sayı aralıklarında hesaplanması başlıca katkılardır.
- Güven zincirlerinde güven ve güvenilirlik yayılımı araştırılmıştır. Güvenilirliğin yayılımı buradaki ana katkıdır.
- Servisler arası güven ve güvenilirlik yayılımı modellenmiştir. Servisler arası sistemlerde güvenilirlik yayılımı bu modelde yapılan başlıca katkıdır.
- Güven Hesaplanması Yazılım Aracı (TAST) adı verilen bir yazılım geliştirilmiştir. Bu yazılım esnekliği sayesinde aynı iş alanında çalışan kuruluşlara kolaylıkla uyarlanabilir. TAST kuruluşların güven değerlerini seçilen zaman aralıklarında hesaplar. TAST rakip kuruluşalara olan güvenin belirlenen zaman aralıklarında kıyaslanmasını sağlar.
- Katkılarımızın uygulanabilirliğini örnekler ve benzetimlerle gösteriyoruz.

Anahtar sözcükler : güven, web, iki-kısımlı çizge, öneri, güven ölçüsü, güven yönetimi, güven iletimi, web üzerinden yapılan anket, güven zinciri.

CONTENTS

Page

CHAPTER ONE
INTRODUCTION

1.1 Motivation

Since the beginning of mankind, trust is an essential basis for human cooperation. A modern society based on the division of labor, people often are willing to rely on others, even though they might face negative consequences. Mutual trust is essential in performing economical transactions in today's world (Hermann, 2003). Today's internet based businesses rely on performing transactions on an adhoc basis with often changing anonymous partners living in other geographical areas with different legal systems. Traditional trust gaining mechanisms cannot be used and new ways to build trust between e-business partners have to be found (Weeks, 2001). In consequence,trust and trust related problems is an emerging research field in the computer science.

Each time we trust someone, we have to put something at risk; our lives, our assets, our properties, and so on. On these occasions, we may use a variety of clues and past experiences to believe these individuals' good intentions towards us and decide on the extent to which we can trust them (Mistzal, 1996). This is the general procedure of trust valuation in daily occasions.

Nowadays, with the development of e-commerce application technologies, a client should look for one service from a large pool of organizations as service providers. In addition to service quality the trustworthiness of an organization is a key factor in selection (Gefen, Srinivasan, & Tractinsky, 2003). This makes trust evaluation a very important issue especially when the client has to select from unknown organizations.

Clients can provide feedback and their trust ratings after completed transactions. Based on the ratings, the trust value of an organization can be evaluated to reflect the quality of services in a certain time period. Trust evaluation approach based on experiences of the former clients is very helpful for the new clients seeking for a trustworthy organization (Mayer, 1995).

Web based surveys. is the fastest and the cheapest way of collecting recommendations of the former clients of the organizations (Budalakoti, DeAngelis, & Barber, 2009).

Trust evaluation approach by using web based survey data collected from their recommenders is the main focus of first-stage of our work in this thesis. Some features that does not exist in various trust models are added in our model.

Propagation of trust over trust chains and in service oriented systems are widely investigated in the following stages of our work. We added some new features in our models of trust propagation.

At the last stage a software tool development has been realized depending on the model of the first stage.

1.2 Contributions

This thesis examines formal trust assessment models. Main contributions of the thesis can be summarized as following:

- A formal model to assess the trust to the organizations in a specified context-set by using web-based survey data is developed. Addition of importance parameter to trust calculations and calculation of trust in real-number intervals by selected confidence probability are the main contributions.
- Trust and confidence propogation in trust chains are investigated. Propogation of confidence is here the main contribution.
- Trust and confidence propogation in service oriented systems are modeled. Propogation of confidence in service-oriented systems is again the main contribution in this model.
- A software tool called Trust Assesment Software Tool (TAST) has been developed. This is a flexible program that can be applied to the organizations working in the same business-field. TAST calculates the trust assessments of

- The organizations in selected time intervals. TAST can make trust assessment comparisons by competitor organizations in selected time intervals.
- We also show the applicability of our contributions by examples and case studies.

1.3 Organization of the Thesis

Thesis has the following structure:

- In Chapter 1, our thesis is introduced.
- In Chapter 2, we provide a detailed overview of trust models and recommender systems.
- In Chapter 3, we introduce a formal graph-based model for trust calculation based on web-based survey data.
- In Chapter 4, trust and confidence propogation in the trust chains and service oriented systems are investigated.
- In Chapter 5, three case studies are given as the application of our models introduced in Chapters 3 and 4.
- In Chapter 6, TAST software is explained in detail.
- In Chapter 7, conclusions and future work are given.

CHAPTER TWO
OVERVIEW OF TRUST MODELING AND RECOMMENDER SYSTEMS

In modern society an individual (or an organization) have limited capacity. We must rely on other people and cooperate with them in our daily life. The interdependence of individuals makes the trust an essential foundation stone of the social and business relations. Trust is a common research field of social sciences and the computer science.

2.1 Trust in Social Sciences

The notion of trust has been frequently used and widely studied in diffeerent disciplines of social sciences such as sociology philosophy, psychology, business management, As a psychologist, Deutsch (1958), has important researches about trust. He defines trust as following:

"An individual may be said to have trust in the occurrence of an event if he expects its occurrence and his expectations lead to behavior which he perceives to have greater negative motivational consequences if the expectation is not confirmed than position motivational consequences if it is confirmed".

Other psychologists Castelfranchi & Falcone (2000) gives a different trust definition:

"Trust is about somebody: it mainly consists of beliefs, evaluations, and expectations about the other actor, his capabilities, selfconfidence, willingness, persistence, morality (and in general motivations), goals and beliefs, etc. Trust in somebody basically is (or better at least includes and is based on) a rich and complex theory of him and of his mind".

As sociologists, McKnight, Cummings & Chervany (1998) gives their trust definition:

"Individuals make trust choices based on rationally derived costs and benefits".

Organizational trust definition is given by a sociologist Coleman (1998).

"The ability of people to work together for common purposes in groups and organizations".

Smith (1998), as a sociologist empasizes the trust as a necessary feature of social work. He defines trust for a modern society following:

"Mutual trust between government and managers and between social workers and service users, represents both a consequence of and a remedy for, uncertainty".

An economist Driscoll (1979), gives the definition of organizational trust:

"Organizational trust is the only significantly useful predictor of overall satisfaction attitudes".

A philosopher Bairer (1986), defines trust as:

"Trust is much easier to maintain than it is to get started and is never hard to destroy".

2.2 Trust in Computer Science

The concept of trust has been widely used and investetigated in computer science. Trust provides many decision making options in different situations. Trust is defined in different manners in computer science by reasearchers like in the field of social sciences.

Starting point of most of today's works related with trust is proposed by Blaze, Feigenbaum, & Lacy (1996). They propose a trust management application named *"Policy Maker Trust Management System".* Policy maker binds public keys to predicates and evaluates proposed actions by interpreting the policy statements and credentials. Depending on the credentials and form of the query it can return either a simple yes/no answer or additional restrictions. Policy maker introduces a general trust management layer. This layer enables the coordination of design policy, contexts and trust relationships.

Jøsang has many proposed researches related with trust modeling. He proposes a new version of probabilistic logic named "*subjective logic"* (Josang, Pope, & Daniel, 2006). Subjective explicitly takes uncertainty about probability values into account. And combines the capability of binary logic to express the structure of argument models with the capacity of probabilities to express degrees of truth of those arguments.

Grandison & Slomon (2000), defines trust for internet applications as following:

" *Trust is the firm belief in the competence of an entity to act dependably, securely and reliably within a specified context* ".

Massa (2006), defines trust in real online systems as: *"The judgement expressed by one user about another user, often directly and explicitly, sometimes indirectly through an evaluation of artifacts produced by thar user or her activity on yhe system".* He also gives categories of trust in online systems according to their similar proporties and common features.

Artz & Gil (2007), proposes that "*trust should refer to mechanisms to verify that the source of information is really who the source claims to be".* Signatures and encryption mechanisms should allow any consumer of information to check the sources of that information.

Mui, Mohtashemi & Fasli (2002) developed a mathematical model to predict feature behaviour of an agent based on past experiences. Their trust definition is as following:

"Trust is a subjective expectation of an agent has about another's future behaviour based on the history of their encounters".

Xiu & Liu (2005), gives a formal definition and analysis of trust in distributed computing environments. Important properties of trust relation, such as reflexivity and conditional transitivity, analyzed and interpreted. Furthermore, for trust relations in "*Role-Based Access Control*" a description is derived.

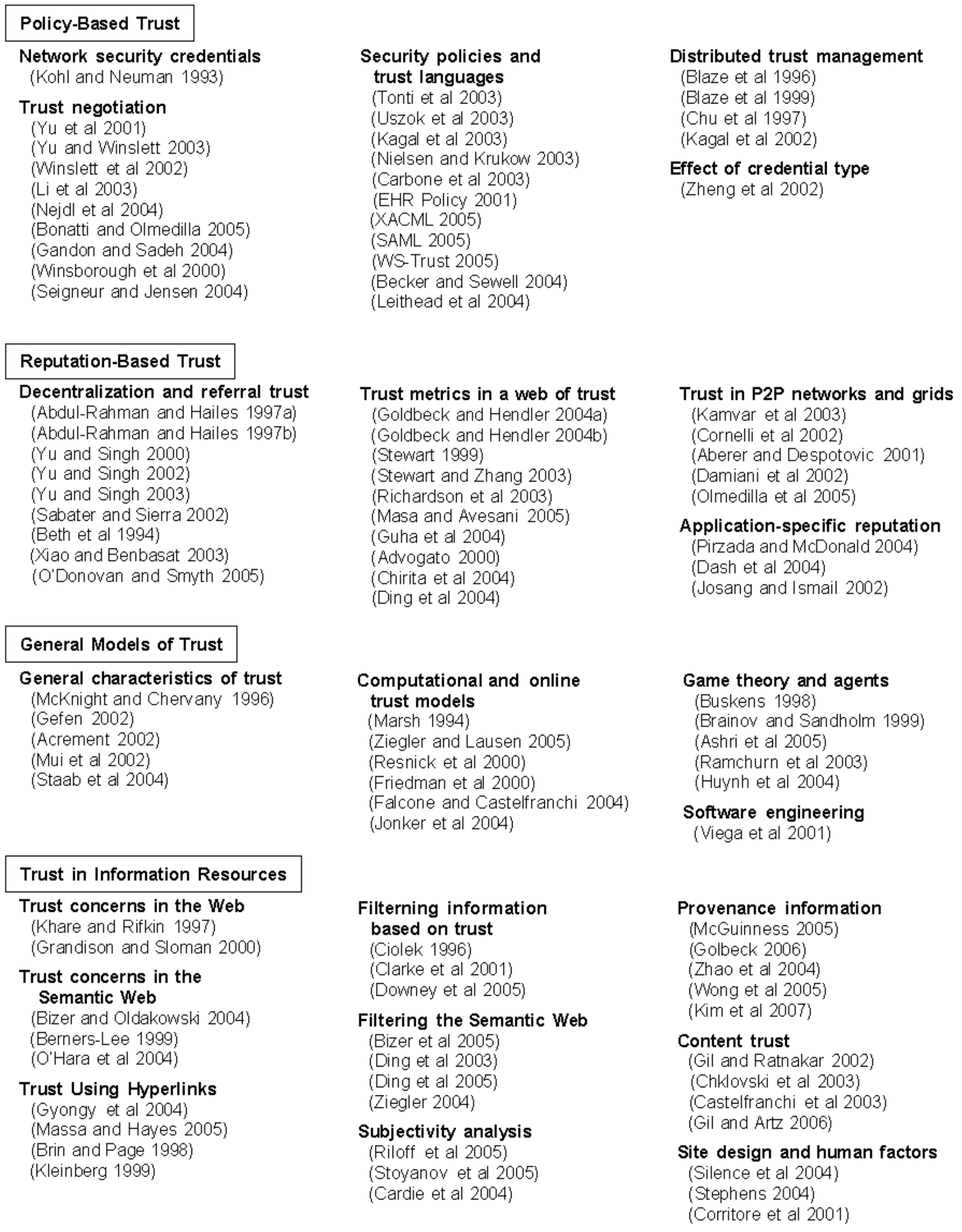
Kuter & Goldbeck (2007), analyse social trust from a computational perspective. They propose a trust inference algorithm called *"SUNNY"*. The algorithm uses a probalistic sampling technique to estimate in trust information for some designated sources.

Li, Huai & Hu (2007), define trust for virtual organizations: *"A virtual organization is of a set of entities, such as resources, services, and users.These entities may belong to different autonomous domains, which collaborate in order to complete certain tasks. VOs have been adopted in many applications such as dynamic enterprises, on-demand computing, on demand services providers, outsourcing business processes, business-to-business collaboration".*

Trust is a complex concept that is difficult to clearly define. There is no consensus in the computer science on what trust is and on what constitutes trust management. Many research scientists recognize its importance and continue to work on trust.

A summary of researches in computer science is given in table 2.1 (Artz & Gil, 2007).

Table 2.1 Summary of trust researches in computer science (Artz, 2007)



Policy-Based Trust		
Network security credentials (Kohl and Neuman 1993) **Trust negotiation** (Yu et al 2001) (Yu and Winslett 2003) (Winslett et al 2002) (Li et al 2003) (Nejdl et al 2004) (Bonatti and Olmedilla 2005) (Gandon and Sadeh 2004) (Winsborough et al 2000) (Seigneur and Jensen 2004)	**Security policies and trust languages** (Tonti et al 2003) (Uszok et al 2003) (Kagal et al 2003) (Nielsen and Krukow 2003) (Carbone et al 2003) (EHR Policy 2001) (XACML 2005) (SAML 2005) (WS-Trust 2005) (Becker and Sewell 2004) (Leithead et al 2004)	**Distributed trust management** (Blaze et al 1996) (Blaze et al 1999) (Chu et al 1997) (Kagal et al 2002) **Effect of credential type** (Zheng et al 2002)

Reputation-Based Trust		
Decentralization and referral trust (Abdul-Rahman and Hailes 1997a) (Abdul-Rahman and Hailes 1997b) (Yu and Singh 2000) (Yu and Singh 2002) (Yu and Singh 2003) (Sabater and Sierra 2002) (Beth et al 1994) (Xiao and Benbasat 2003) (O'Donovan and Smyth 2005)	**Trust metrics in a web of trust** (Goldbeck and Hendler 2004a) (Goldbeck and Hendler 2004b) (Stewart 1999) (Stewart and Zhang 2003) (Richardson et al 2003) (Masa and Avesani 2005) (Guha et al 2004) (Advogato 2000) (Chirita et al 2004) (Ding et al 2004)	**Trust in P2P networks and grids** (Kamvar et al 2003) (Cornelli et al 2002) (Aberer and Despotovic 2001) (Damiani et al 2002) (Olmedilla et al 2005) **Application-specific reputation** (Pirzada and McDonald 2004) (Dash et al 2004) (Josang and Ismail 2002)

General Models of Trust		
General characteristics of trust (McKnight and Chervany 1996) (Gefen 2002) (Acrement 2002) (Mui et al 2002) (Staab et al 2004)	**Computational and online trust models** (Marsh 1994) (Ziegler and Lausen 2005) (Resnick et al 2000) (Friedman et al 2000) (Falcone and Castelfranchi 2004) (Jonker et al 2004)	**Game theory and agents** (Buskens 1998) (Brainov and Sandholm 1999) (Ashri et al 2005) (Ramchurn et al 2003) (Huynh et al 2004) **Software engineering** (Viega et al 2001)

Trust in Information Resources		
Trust concerns in the Web (Khare and Rifkin 1997) (Grandison and Sloman 2000) **Trust concerns in the Semantic Web** (Bizer and Oldakowski 2004) (Berners-Lee 1999) (O'Hara et al 2004) **Trust Using Hyperlinks** (Gyongy et al 2004) (Massa and Hayes 2005) (Brin and Page 1998) (Kleinberg 1999)	**Filterning information based on trust** (Ciolek 1996) (Clarke et al 2001) (Downey et al 2005) **Filtering the Semantic Web** (Bizer et al 2005) (Ding et al 2003) (Ding et al 2005) (Ziegler 2004) **Subjectivity analysis** (Riloff et al 2005) (Stoyanov et al 2005) (Cardie et al 2004)	**Provenance information** (McGuinness 2005) (Golbeck 2006) (Zhao et al 2004) (Wong et al 2005) (Kim et al 2007) **Content trust** (Gil and Ratnakar 2002) (Chklovski et al 2003) (Castelfranchi et al 2003) (Gil and Artz 2006) **Site design and human factors** (Silence et al 2004) (Stephens 2004) (Corritore et al 2001)

2.3 Properties of Trust

Trust relationships between entities may be in various patterns (Oliviera, Pelusoa, & Romano, 2008).

- One to one

- One to many
- Many to many

Trust of one entitity to another is always subjective. That means trust depends on personal opinion (Josang, Keser, & Dimitrakos, 2005). Personal opinions are formed by some factors and evicendence and may change person to person.

Trust always depens on a context. If context changes trust also changes (Ma & Orgun, 2006). Therefore the context on which trust relation is based on must be clearly defined.

Trust is directed. That means trust is not symmetric (Carroll, Bizer, Hayes, & Stickler, 2005). If a person trusts some the other person does not necessarily trust to him/her.

Trust values are used to represent the degrees of trust relationships. Trust values enables us to model and analyze the trust based systems (Lang, 2010). Trust is a measurable blief.

Trust changes with time (Bahtiyar, Cihan, & Caglayan, 2010). Trust value changes with time by the factors events, actions, and etc. Dynamism of trust forces trust management systems to hae properties like learning and reasoning solutions (Yan, 2007).

Trust is transferable, but does not have relational transitivity (Bargh, Jansen, & Smith, 1998). Trust can be transfered under certain conditions.

In summary, the number of trust properies vary from one trust system to another one. Moreover in the literature there some other properties that are defined for trust.

2.4 General Trust Models

Trust models generally determine the degree of trust between two entities. The first trust model is the direct trust model (Sun, Han, & Liu, 2008). Trust between entities is established depending on the previous direct interactions between entities. There is no trust propogation.

Second trust model is transitive trust model. In this model trust is transmitted entities. This model is also called indirect trust model. Transitivity property is based on propogation of trust (Andert, Wakefield, & Weise, 2002). Two important factors must be considered for trust transitivity. First factor is how and when to collect trust information (Biskup, Hielser, & Wortmann, 2008). Second factor is how to calculate trust values for propogation. The advantage of trust transitivity is to connect different entities that share similar credentials (Hang, Wang, & Singh, 2008).

Trust is not always transitive. There are some situations like some entities may not use the information obtained for one context which is used by other entities (Burgess, Canright, & Monsen, 2004).

2.5 Trust Representation Models

Generally, entities express their trust as percentage and less commonly with an absolute value. However, depending on the nature of relations between entities various ways to represent the value of trust are used.

- *Discrete Trust Models:* Expressing trust in discrete data is easier than using the probability statements. It would be simpler to say that an entity is *usually trusted* rather than expressing such statement as a percentage like trusted in 60% of cases . In a binary scale for the expression of trust, an entity declares its trust in another as the positive value of 1, or distrust by as the negative value -1. The zero indicates that there is no declared trust relationship between the two entities (Orgun & Liu, 2006).

- *Probabilistic Trust Models:* The main purpose of expressing trust with probabilities is to apply methods based on probability calculus. Probabilistic models use advanced robust statistical methods such as Bayesian approaches or Markov chains (Ben-Gal, Ruggeri, Faltin, & Kenett, 2007). Probabilistic calculation methods can be used either in a system of continous or discrete values.
- *Belief Models:* In *Belief Models*, trust is a continuous value composed of trust distrust and the uncertainty. The sum of these three values is equal to 1. The Belief Models proposed by Josang, Mollerud, & Chung (2001). Josang's model combines trust and distrust to represent the belief of an entity on another entity and can be be less than 1. The difference between 1 and the belief value is the uncertainity value.
- *Fuzzy Models:* Fuzzy logic is suitable for trust evaluation because it is possible to handle conflicting trust values by using fuzzy linguistic expression (e.g. low, medium, high). Using fuzzy linguistic expression makes easier to assign trust values for users (Chen, Bu, Zhang, & Zhu, 2005).

Above, main computational models of trust and reputation have been developed are given. Independendent of the chosen model, the requirements expected from the model can be summarized as follows (Liu, Ozols, & Orgun, 2005).

- The model must provide a trust metric that represents a level of trust in an agent. Such a metric allows comparisons between agents so that one agent can be accepted as more trustworthy than another. The model must be able to provide a trust metric in the presence or absence of personal experience.
- The model must reflect an individual's confidence in its level of trust for another agent. This is necessary because an agent can determine the degree of influence of the trust metric on the decision about whether to interact with another individual. Higher confidence means a greater influence on the decision-making process, and lower confidence means less influence.
- The model should handle bootstrapping. That means, when neither the truster or its opinion providers have previous experience with a trustee. The

truster can still assess the trustee based on other information it may have available.

2.6 Trust Related Terms

Trust definitions in computer science are different in each context. Different models use different terms related to trust (Neisse, Wegdam, & Sinderen, 2006). In this section we will explain the trust related terms frequently used in literature.

- *Trust*: Trust is "the belief in the competence of an entity to act dependably, securely and reliably within a specifed context" (Grandison & Sloman, 2000).
- *Entity:* An entity is a unit which is aware of other entity's trustworthiness. It also has the ability to decide under which conditions to set up interactions with other entities (Rasmusson & Janson, 1996). An entity can be:
    - a person
    - an agent
    - a host
    - a device
    - a process
    - a service
- *Truster (or relying party):* Truster is an entity that trusts another entity.
- *Trustee (or relied party):* Trustee is an entity that is trusted by another entity.
- *Trust Relationship*: A trust relationship can only exist between two entities. It reflects the truster's opinion about the trustee's trustworthiness. A trust relationship is uni-directional. If entity A trusts entity B and entity B trusts entity A, each trust relationship will be considered separately. A trust relationship is dynamic and may change over time (Jeffrey, 2004).
- *Belief:* Belief is an entity's opinion about something to accept it as truth. Belief is subjective because it changes from entity to another entity about the same case (Josang, 2002).
- *Reputation:* Reputation is considered as a collective measure of trustworthiness based on ratings (Massa, 2003).

- *Context:*Trust is always based on a context. Dey (2001) defines the context as *"any information that can be used to characterise the situation of entities. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves".* Contexts are divided into *direct* or *recommended* contexts to reflect the nature of the trustee in the relationship. Context is sometimes called as *trust scope*.
- *Experience:* An experience is obtained as result of interacting with an entity. Experience shows how trustworthy the trustee behaved in that interaction. Experiences are divided into *direct* or *recommended* experiences to reflect the nature of the trustee in the relationship (Josang, Ismail, & Boyd, 2007).
- *Direct Trust:* Direct trust is based on truster's own experiences about trustee. No recommendations are considered (Sebater i Mir, 2003).
- *Confidence:* Confidence represents the level of truster's trust on trustee. Confidence can be considered as a metric that represents the accuracy of the trust value calculated. Higher confidence means a greater impact on the decision-making process, and lower confidence means less impact. Purser (2001) gives a definition for confidence as follows.

  *"The associated confidence level**:** The degree of confidence that the trusted entity will not violate the trust. He models this as 'high','medium' or 'low".*

  Another definition of confidence is given by Zejda (2010) as *"the accuracy or the quality of trust where high confidence is more useful in making trust definitions".*

  Confidence in relation with trust is used as a confidence level that helps to use statistical properties of trust. In statistics a confidence level is generally described as a confidence interval or confidence bound that is an interval estimate of a population parameter. Reliability of an estimate is represented by confidence intervals (Gentle, Hardle, & Mori, 2004).

- *Recommendation:* Recommendation is the opinion of a third party entity about the trustee’s trustworthiness. Recommendation is sometimes called as *referral* or *indirect trust* (Carbone, Nielsen, & Sassone, 2003).
- *Trust Transitivity:* Trust is conditionally transferable. Information about trust can be transmitted or received by means of a chain of recommendations. The conditions are bound to the context and the truster’s objective factors (Ray & Chakraborty, 2009).
- *Trust Value:* Trust Value indicates the strength of the trust relationship between the truster and the trustee (Trcek, 2009).
- *Trust Metric:* Trust Metric defines the method of calculation of some trust value based on direct and indirect trust (Raya, Papadimitratos, Gligor, & Hubaux, 2008).
- *Trust Treshold:* The trust threshold is a trust value established by the truster. All trustees whose trust values are above the threshold are trusted by truster. Otherwise they are untrusted (Zhou & Hwang, 2007).
- *Inferred Trust:* Inferred trust is the value of the referral trust (or recommendation) obtained over a trust chain (Guha, Kumar, Raghavan, & Tomkins, 2004).
- *Time:* An important element to a trust relationship is its time component. Trust of the trustor in the trust target might be quite different with time passing.

2.7 Recommender Systems

Recommender systems are emerging all around the world using reputation-aware systems. People use recommender systems to advice other people movies, books, songs, cars etc. The information that other people provide may come from explicitly from ratings, tags, reviews, or implicitly from how they spend their money and time.The information obtained can be used to select, fitler, or sort items. The recommendations may be personilized to the preferences of different users (Yolum, 2003).

In general, recommender systems are based on one of three methods (Scahafer, Konstan, & Riedl, 1999).

- Content filtering.
- Colloborative filtering.
- Hybrid methods

Content filtering approach creates a profile for each product or customer. These profiles describe their nature (Huang, Chung, & Chen, 2004). For example, a car profile could include its features like its speed, its engine power, its fuel consumption, available colors etc. Customer profiles about their car model are collected by means of surveys. Surveys include a suitable set of questions about factors affecting their car prefernces. Personal questions like their gender, age, education address, phone etc. may be included (Koren, Bell, & Volinsky, 2009). When enough information is collected to match user and car profiles a software can be used. Content filtering based methods require gathering information directly from users might not be easy (Cremonesi, Garzotto, Negro, Papadapoulos, & Turrin, 2011).

The alternative method is called collaborative filtering (Schafer, Frankowski, Herlocker, & Sen, 2007). Collobarative filtering relies on the past behaviour of the customers. Examples can be customer’s previous shoppings, types of products bought, choice of brands etc.

Collaborative filtering is more successful to analyse product customer relationships (Hu, Koren, & Volinsky, 2008). In case of new products to new customer relationships content filtering is more successful. Hybrid systems are a combination of these two.

Compared to similar works, our research can be named as a specialized content filtering method focusing on set of contexts describing activities of organizations.

Application of web-based surveys simplfy the difficulty of collecting customer satisfaction feedback information.

2.8 Graphs

In mathematics and computer science, graph theory is the study of graphs. Mathematical structures used to model pairwise relations between objects from a certain collection. A graph in this context refers to a collection of vertices or nodes and a collection of edges that connect pairs of vertices. A graph may be undirected, meaning that there is no distinction between the two vertices associated with each edge, or its edges may be directed from one vertex to another which is defined by Knobloch, E., Leibniz, & Euler (1991).

A graph G consists of two types of elements, namely vertices and edges. Every edge has two endpoints in the set of vertices, and is said to connect or join the two endpoints. An edge can thus be defined as a set of two vertices (or an ordered pair, in the case of a directed graph). Alternative models of graph exist; e.g., a graph may be thought of as a Boolean binary function over the set of vertices or as a square (0,1) matrix. A vertex (basic element) is simply drawn as a node or a dot. The vertex set of G is usually denoted by V(G), or V when there is no danger of confusion. The order of a graph is the number of its vertices, i.e. |V(G)|. An edge (a set of two elements) is drawn as a line connecting two vertices, called endvertices, or endpoints. An edge with endvertices x and y is denoted by xy (without any symbol in between). The edge set of G is usually denoted by E(G), or E when there is no danger of confusion. The size of a graph is the number of its edges, i.e. |E(G)| defined by Diesel (2000).

A graph is a pair G graph = (V;E) of sets satisfying E $\subseteq [V]^2$; thus, the elements of E are 2-element subsets of V. The elements of V are the vertex vertices (or nodes, or points) of the graph G, the elements of E are its edge edges (or lines). The usual way to picture a graph is by drawing a dot for each vertex and joining two of these dots by a line if the corresponding two vertices form an edge. Just how these dots and lines are drawn is considered irrelevant: all that matters is the information which pairs of vertices form an edge and which do not.

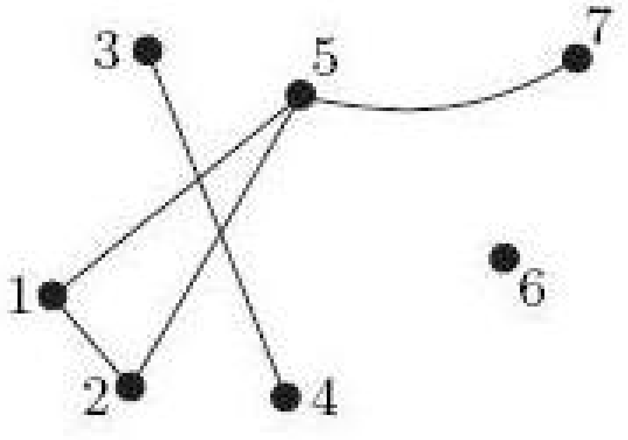


Figure 2.1 The graph on V = {1, . . . , 7} with edge set
E = {{1, 2}, {1, 5}, {2, 5}, {3, 4}, {5, 7}}, (Diesel,2000)

A graph with vertex set V is said to be a graph on V. The vertex set of a graph G is referred to as V(G), its edge set as E(G). The number of vertices of a graph G is its order, written as |G|**;** its number of edges is denoted by ||G||. Graphs are finite or infinite according to their order.

A loop is an edge whose endvertices are the same vertex. A link has two distinct endvertices. An edge is multiple if there is another edge with the same endvertices; otherwise it is simple. The multiplicity of an edge is the number of multiple edges sharing the same endvertices; the multiplicity of a graph, the maximum multiplicity of its edges. A graph is a simple graph if it has no multiple edges or loops, a multigraph if it has multiple edges, but no loops, and a multigraph or pseudograph if it contains both multiple edges and loops. When stated without any qualification, a graph is almost always assumed to be simpleone has to judge from the context. Graph labeling usually refers to the assignment of unique labels (usually natural numbers) to the edges and vertices of a graph. Graphs with labeled edges or vertices are known as labeled, those without as unlabeled. More specifically, graphs with labeled vertices only are vertex-labeled, those with labeled edges only are edge-labeled defined by Knobloch and et al. (1991).

A subgraph of a graph G is a graph whose vertex set is a subset of that of G, and whose adjacency relation is a subset of that of G restricted to this subset. In the other direction, a supergraph of a graph G is a graph of which G is a subgraph. It is said a graph G contains another graph H if some subgraph of G is H or is isomorphic to H. A subgraph H is a spanning subgraph, or factor, of a graph G if it has the same vertex set as G. It is said H spans G.

2.8.1 Colored Graphs

A colored graph is a complete graph in which a color has been assigned to each edge, and a colorful cycle is a cycle in which each edge has a different color (Ball, Pultr, & Vojtechovsky, 2007). Gallai graphs, are the graphs in which every triangle has edges of exactly two colors. They can be iteratively built up from three simple colored graphs, having 2, 4, and 5 vertices, respectively. An edge coloring of a graph is an assignment of *colors* to the edges of the graph so that no two adjacent edges have the same color. The edge-coloring problem asks whether it is possible to color a given graph using at most *n* colors. The minimum required number of colors for a graph is called the chromatic index. For example, if a graph can be colored by three colors but cannot be colored by two colors, it has a chromatic index three. The smallest number of colors needed in a proper edge coloring of a graph *G* is the *chromatic index.*

An edge coloring of a graph, when mentioned without any qualification, is always assumed to be a proper coloring of the edges, that means no two adjacent edges are assigned the same color. *Adjacent* means sharing a common vertex. A proper edge coloring with *k* colors is called a proper k-edge-coloring and is equivalent to the problem of partitioning the edge set into k matchings. A graph that can be assigned a proper k-edge-coloring is k-edge-colorable.

2.8.2 Bipartite Graphs

In the mathematical field of graph theory a *bipartite graph* (*or bigraph*) is a graph vertices can be divided into two disjoint sets *U* and *V* such that every edge connects a vertex in *U* to one in *V* (Gross, & Yellen, 2003). That means, *U* and *V* are independent sets. A bipartite graph is a graph that does not contain any odd-length cycles.The two sets *U* and *V* may be thought of as a coloring of the he graph with two colors. If we color all nodes in *U* blue, and all nodes in *V* green, each edge has endpoints of differing colors. Such a coloring is impossible in the case of a nonbipartite graph. For example in the case of a triangle, after one node is colored blue and another green, the third vertex of the triangle is connected to vertices of both colors, prevents it from being assigned either color. A simple bipartite graph is shown in figure 2.2.

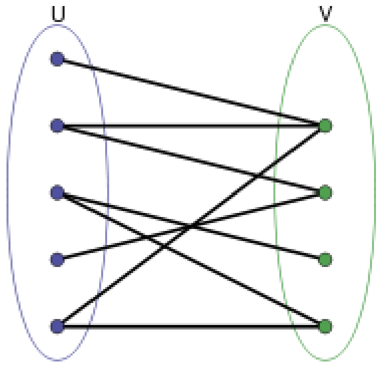


Figure 2.2 A simple bi-partite graph, (Diestel,2000)

If a bipartite graph is connected, its bipartition is defined by the parity of the distances from any arbitrarily chosen vertex *v*. One subset consists of the vertices at even distance to *v* and the other subset consists of the vertices at odd distance to *v*. So, one may efficiently test whether a graph is bipartite by using this parity technique to assign vertices to the two subsets *U* and *V*, separately within each connected component of the graph Then examine each edge to verify that it has endpoints assigned to different subsets. *G* = (*U*, *V*, *E*) denotes a bipartite graph whose partitions has the parts *U* and *V*. If |*U*| =|*V*|, the two subsets have equal cardinality, then *G* is called a *balanced bipartite* graph.

Some properties of bipartite graphs can be summarized as follows:

- A graph is bipartite if and onl if it does not contain an odd cycle. Therefore, a bipartite graph cannot contain a clique of size 3 or more.
- A graph is bipartite if and only if it is 2-colorable, (i.e. its chromatic number is less than or equal to 2).
- The size of minimum vertex coveris s equal to the size of the maximum mathing.(König’s theorem)
- The size of the maximum independent set plus the size of the maximum matching is equal to the number of vertices.
- For a connected bipartite graph the size of the minimum edge cover is equal to the size of the maximum independent set.
- For a connected bipartite graph the size of the minimum edge cover plus the size of the minimum vertex cover is equal to the number of vertices.
- Every bipartite graph is a perfect graph.
- The spectrum of a graph is symmetric if and only if it's a bipartite graph.

2.9 Confidence Interval and Confidence Level

The *confidence interval* (also called margin of error) is the plus-or-minus figure usually reported in newspaper or television opinion poll results. For example, if you use a confidence interval of 4 and 47% percent of your sample picks an answer you can be "sure" that if you had asked the question of the entire relevant population between 43% (47-4) and 51% (47+4) would have picked that answer (Neuman, 2000).

The *confidence level* tells you how sure you can be. It is expressed as a percentage and represents how often the true percentage of the population who would pick an answer lies within the confidence interval. The 95% confidence level means you can be 95% certain; the 99% confidence level means you can be 99% certain. Most researchers use the 95% confidence level (Neuman, 2000).

When you put the confidence level and the confidence interval together, you can say that you are 95% sure that the true percentage of the population is between 43% and 51%. The wider the confidence interval you are willing to accept, the more certain you can be that the whole population answers would be within that range.

For example, if you asked a sample of 1000 people in a city which brand of cola they preferred, and 60% said brand A, you can be very certain that between 40 and 80% of all the people in the city actually do prefer that brand, but you cannot be so sure that between 59 and 61% of the people in the city prefer the brand.

2.9.1 Factors that Affect Confidence Intervals

There are three factors that determine the size of the confidence interval for a given confidence level.

- Sample size
- Percentage
- Population size

2.9.2 Sample Size

The larger your sample size, the more sure you can be that their answers truly reflect the population. This indicates that for a given confidence level, the larger your sample size, the smaller your confidence interval. However, the relationship is not linear, Doubling the sample size does not halve the confidence interval (Hines, Montgomery, Goldsman, & Borror, 2003).

2.9.3 Percentage

Your accuracy also depends on the percentage of your sample that picks a particular answer. If 99% of your sample said "Yes" and 1% said "No," the chances of error are remote, irrespective of sample size. However, if the percentages are 51% and 49% the chances of error are much greater. It is easier to be sure of extreme answers than of middle-of-the-road ones.

When determining the sample size needed for a given level of accuracy you must use the worst case percentage (50%). You should also use this percentage if you want to determine a general level of accuracy for a sample you already have. To determine the confidence interval for a specific answer your sample has given, you can use the percentage picking that answer and get a smaller interval (Hines, Montgomery, Goldsman, & Borror, 2003).

2.9.4 Population Size

How many people are there in the group your sample represents? This may be the number of people in a city you are studying, the number of people who buy new cars, etc. Often you may not know the exact population size. This is not a problem. The mathematics of probability proves the size of the population is irrelevant unless the size of the sample exceeds a few percent of the total population you are examining. This means that a sample of 500 people is equally useful in examining the opinions of a state of 15,000,000 as it would a city of 100,000. For this reason, The survey system ignores the population size when it is large or unknown. Population size is only likely to be a factor when you work with a relatively small and known group of people.

The confidence interval calculations assume you have a genuine random sample of the relevant population. If your sample is not truly random, you cannot rely on the intervals. Non-random samples usually result from some flaw in the sampling procedure. An example of such a flaw is to only call people during the day and miss almost everyone who works. For most purposes, the non-working population cannot be assumed to accurately represent the entire working and non-working population (Hines, Montgomery, Goldsman, & Borror, 2003).

2.9.5 Normal Distribution

The normal curve is a bell-shaped, symmetrical graph with an infinitely long base. The mean, median, and mode are all located at the center as shown in figure 2.3.

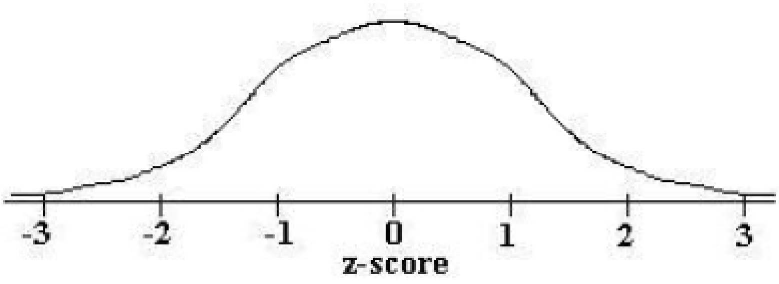


Figure 2.3 Normal distribution, (Diestel,2000)

A value is said to be normally distributed if its histogram is the shape of the normal curve. The probability that a normally distributed value will fall between the mean and some z-score z is the area under the curve from 0 to z as shown in figure 2.4. Areas from mean to z-score are shown in table 2.2.

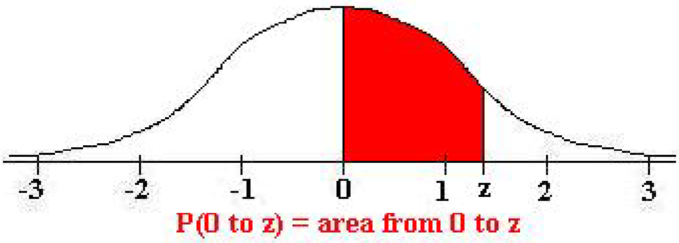
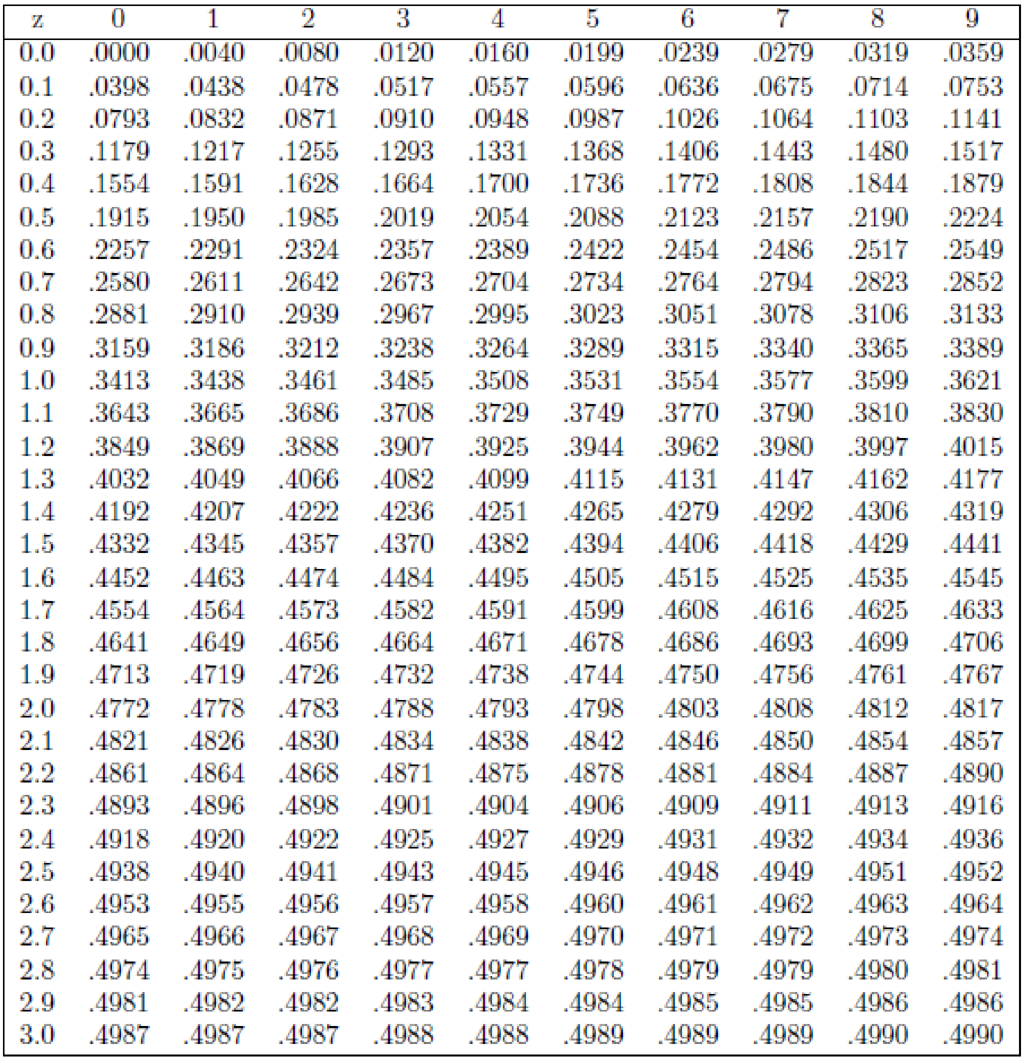


Figure 2.4 Z is the area under the curve from 0 to z, (Diestel,2000)

Table 2.2 Areas from the mean to z-score, (Diestel,2000)



z	0	1	2	3	4	5	6	7	8	9
0.0	.0000	.0040	.0080	.0120	.0160	.0199	.0239	.0279	.0319	.0359
0.1	.0398	.0438	.0478	.0517	.0557	.0596	.0636	.0675	.0714	.0753
0.2	.0793	.0832	.0871	.0910	.0948	.0987	.1026	.1064	.1103	.1141
0.3	.1179	.1217	.1255	.1293	.1331	.1368	.1406	.1443	.1480	.1517
0.4	.1554	.1591	.1628	.1664	.1700	.1736	.1772	.1808	.1844	.1879
0.5	.1915	.1950	.1985	.2019	.2054	.2088	.2123	.2157	.2190	.2224
0.6	.2257	.2291	.2324	.2357	.2389	.2422	.2454	.2486	.2517	.2549
0.7	.2580	.2611	.2642	.2673	.2704	.2734	.2764	.2794	.2823	.2852
0.8	.2881	.2910	.2939	.2967	.2995	.3023	.3051	.3078	.3106	.3133
0.9	.3159	.3186	.3212	.3238	.3264	.3289	.3315	.3340	.3365	.3389
1.0	.3413	.3438	.3461	.3485	.3508	.3531	.3554	.3577	.3599	.3621
1.1	.3643	.3665	.3686	.3708	.3729	.3749	.3770	.3790	.3810	.3830
1.2	.3849	.3869	.3888	.3907	.3925	.3944	.3962	.3980	.3997	.4015
1.3	.4032	.4049	.4066	.4082	.4099	.4115	.4131	.4147	.4162	.4177
1.4	.4192	.4207	.4222	.4236	.4251	.4265	.4279	.4292	.4306	.4319
1.5	.4332	.4345	.4357	.4370	.4382	.4394	.4406	.4418	.4429	.4441
1.6	.4452	.4463	.4474	.4484	.4495	.4505	.4515	.4525	.4535	.4545
1.7	.4554	.4564	.4573	.4582	.4591	.4599	.4608	.4616	.4625	.4633
1.8	.4641	.4649	.4656	.4664	.4671	.4678	.4686	.4693	.4699	.4706
1.9	.4713	.4719	.4726	.4732	.4738	.4744	.4750	.4756	.4761	.4767
2.0	.4772	.4778	.4783	.4788	.4793	.4798	.4803	.4808	.4812	.4817
2.1	.4821	.4826	.4830	.4834	.4838	.4842	.4846	.4850	.4854	.4857
2.2	.4861	.4864	.4868	.4871	.4875	.4878	.4881	.4884	.4887	.4890
2.3	.4893	.4896	.4898	.4901	.4904	.4906	.4909	.4911	.4913	.4916
2.4	.4918	.4920	.4922	.4925	.4927	.4929	.4931	.4932	.4934	.4936
2.5	.4938	.4940	.4941	.4943	.4945	.4946	.4948	.4949	.4951	.4952
2.6	.4953	.4955	.4956	.4957	.4958	.4960	.4961	.4962	.4963	.4964
2.7	.4965	.4966	.4967	.4968	.4969	.4970	.4971	.4972	.4973	.4974
2.8	.4974	.4975	.4976	.4977	.4977	.4978	.4979	.4979	.4980	.4981
2.9	.4981	.4982	.4982	.4983	.4984	.4984	.4985	.4985	.4986	.4986
3.0	.4987	.4987	.4987	.4988	.4988	.4989	.4989	.4989	.4990	.4990

2.9.6 Central Limit Theorem

Start with a population with a given mean μ and standard deviation σ. Take samples of size n, where n is a sufficiently large (generally at least 30) number, and compute the mean of each sample (Diestel,2000).

- The set of all sample means will be approximately normally distributed.
- The mean of the set of samples will equal μ, the mean of the population .

- The standard deviation $\sigma_{\bar{x}}$, of the set of sample means will be approximately $\frac{\sigma}{\sqrt{n}}$.

2.9.7 Linear Transformations

A linear transformation of a data set is one where each element is increased by or multiplied by a constant. This affects the mean, the standard deviation, in different ways (Diestel,2000).

- Addition: If a constant c is added to each member of a set, the mean will be c more than it was before the constant was added; the standard deviation and variance will not be affected.
- Multiplication: Another type of transformation is multiplication. If each member of a set is multiplied by a constant c, then the mean will be c times its valuebefore the constant was multiplied; the standard deviation will be |c| times its value before the constant was multiplied.

2.10 Other Works on Trust Assesment and Models

Other related important works are summarized in the following pharagraphs.

Hermann (2006) proposed a software toll named cTLA which is a linear time temporal logic describing properties of state transition systems by means of often lengthy and complex canonical formulas. CTLA is based on developed by Lamport (2002). In contrast to TLA, cTLA omits the canonical parts of TLA formulas. CTLA is oriented at programming languages and introduces the notion of processes. A specification is structured into modular definitions of process type. An instantiation of a process type introduces the notion of process and systems or subsystems are defined as the composition of concurrent process descriptions. CTLA allows to carry out deduction proofs that an implementation of a trust management system fulfills a trust model and particular trust properties. Different from other formalisms in the literature cTLA takes relevant aspects of trust including time and context. However, trust evaluation of the trust value is done based on only reputation. Herrmann models reputation based trust as a decaying value, since recent information about an entity's reputation affects the level of trust to that entity more than past information. For this purpose, a simple decay function is introduced. In cTLA computation of the trust values is based on Jonsang's subjective logic.

Orgun & Liu (2006) describe agent as being a person, a computer, a handheld device or some other entity. Agents should gain their beliefs regarding whether messages they received are reliable based on their trust in the security mechanisms of a system. Therefore, it is important to provide a formal method for specifying the trust that agents have in the security mechanisms of the system. So it will be possible as to support reasoning about agent beliefs as well as the security properties that the system may satisfy. It is clear that any logical system modeling active agents should be a combined system of logics of knowledge, belief, time and context.

Liu, Ozols, & Orgun, (2005). propose *Typed Modal Logic* as an extension of first order logic with typed variables and modal operators to express beliefs of agents. Based on TML, system-specific theories of trust can be constructed, and they provide a basis for analysing and reasoning about trust in particular environments and systems. TML seems to be more suitable to express static properties of trust. Trust can therefore be developed over time as the outcome of a series of confirming observations . An agent may lose its trust or gain new trust at any moment in time due to some reasons such as recommendations from other agents. Without the introduction of a temporal dimension, TML is unable to express the dynamics of trust.In order to form TML+, atoms of TLC are allowed to be substituted by the formulas of TML. However, substitution of TML atoms by TLC formulas is not allowed. This causes some restrictions in the resulting logic such as only being able to reason about the temporal aspects of agent beliefs. In order to interpret a formula written in TML+, one needs a time reference. After having done the mapping of the formula to a specific moment in time, the meaning of the remaining subformula can be decided by an association to the model. This an advantage of the resulting logic TML+ as its semantics is understandable. The disadvantage of this model is that it is based on binary trust values meaning either trust or no trust.

Duterte (1995) proposes a model based on ITL and DC which are first order logics. They support the expressions with quantitative real-time requirements. These two logics have in common the presence of a binary modal operator called the "chop" operator denoted by ";". Chop operator performs the action of splitting a time interval in two parts. His model constructs a complete and sound proof system for classes of ITL each of which make different assumptions about time. He claims that complete axiomatic systems for different classes of ITL can be obtained by using the construction presented in his paper.

Moszkowski (2007) proposes a propositional version of Interval Temporal Logic (ITL) which named as PITL. It is a natural generalization of PTL and includes operators for reasoning about periods of time and sequential composition. Versions of PTL with finite time and infinite time are both considered. One of benefits of the framework is the ability to systematically reduce infinite-time reasoning to finite-time reasoning. The treatment of PTL with the operators until and past time naturally reduces the effort spent. The interval-oriented methodology differs from other analyses of PTL which typically use sets of formulas and sequences of such sets for canonical models. Instead, models are represented as time intervals expressible in PITL.The analysis furthermore relates larger intervals with smaller ones. Being an interval-based formalism, PITL is well suited for sequentially combining and decomposing the relevant formulas. Existence of bounded models with periodic suffixes for PTL formulas which are satisfiable in infinite time. Decision procedures based on binary decision diagrams and exploit some links with finite-state automata. Beyond the specific issues involving PTL, PITL is a significant application of ITL and interval-based reasoning and illustrates a general approach to formally reasoning about sequential and parallel behaviour in discrete linear time.

Aziz, Singhal, & Balarin (1995) propose pCTL which is a probabilistic variant of Computational Tree Logic. In their work, the authors show that pCTL can be interpreted over discrete Markov processes. They define a bi-simulation relation on finite Markov processes and show that Markov processes are sound and complete with respect to pCTL. Generalized Discrete Markov Processes, which is an extension of this model can be used for formalization of the trust concept. The reason for this is that generalized Markov Processes can be used for modeling systems where transition probabilities are not completely specified.

Bertino, Ferrari, & Squicciarini (2004) propose X-TNL as a XML based language. It is developed for specifying Trust-X certificates and disclosure policies. The use of an XML formalism for specifying credentials facilitates credential submission and distribution, analysis and verification by use of a standard query language such as XQuery. X-TNL certificates are the means to convey information about the profile of the parties involved in the negotiation. A certificate can be either a credential or a declaration. A credential is a set of properties of a party certified by a CA and digitally signed by the issuer, according to the Standard defined by W3C for XML. To enforce both trust and efficient negotiations, X-TNL supports the notion of trust ticket. Trust tickets are a powerful means to reduce as much as possible the number of certificates and policies that need to be exchanged during negotiations.Trust tickets are generated by each of the involved parties at the end of a successful negotiation and issued to the corresponding counterpart. Like conventional certificates, trust tickets are locally stored by their owners into their X-Profile, in a specific data set.

Esfendiari & Chandrasekharan (2001) emphesize the importance of e-commerce and propose methods to determine the credentials of the buyer or the seller before initiating a commercial transaction. They explore different Trust Acquisition Mechanisms, by describing different ways to calculate and update trust.These are: Trust Acquisition by Observation, Trust Acquisition by Interaction, Trust Acquisition Using Institutions. They propose to use a directed graph for trust evaluation. In a multi-agent, distributed, setting, where the graph's edge values are not centrally known, the problem of calculation of the trust interval becomes equivalent to the problem of routing in a communication network. Since the trust is only weakly transitive , their propagation model takes into account the decrease of trust along the chain. In an optimistic setting they propose that the agent can use the max value as a decision threshold, whereas in a pessimistic setting the agent can use the min value. They also note that another problem with propagation is that the notion of trust might vary for each agent-agent relationship. Agents might build trust for different aspects of their acquaintances, for example assign trust for a particular task. Therefore they need to have colored edges, with a color per task or type of trust. And they would have a "multi-colored" edge for "general" trust. Trust would only propagate through edges of the same color.

Trcek (2009) introduced trust graphs to study propagation of trust in social interactions. The links of trust graphs are directed and weighted accordingly. If a link denotes the trust attitude of agent A towards agent B, the link is directed from A to B. Because graphs can be equally presented with matrices . Trust matrix operations are not the same as those in ordinary linear algebra. Rows represent a certain agent's trust towards other agents, while columns (or trust vectors) represent trust of the community related to a particular agent. Further, an interesting case with this algebra for computing environments is the possibility of including trust of technological components or services.

Yao, Shin, Tamassia. & Winsborough (2005) propose an interactive visualization framework for the automated trust negotiation (ATN) protocol and they have implemented a prototype of the visualizer in Java.This framework provides capabilities to perform the interactive visualization of an ATN session, display credentials and policies, analyze the relations of negotiated components, and refine access control policies and negotiation strategies. They give examples of the visualization of ATN sessions and demonstrate the interactive features of the visualizer for the incremental construction of a trust target graph (TTG).

Ma & Orgun (2006) propose a formal approach to a revising theory of trust, which includes techniques for modeling trust changes and theory changes. They define a method for computing the new trust state from the old one and its change, and a method to obtain the theory change corresponding to a given trust change. Since trust changes dynamically, to express the dynamics of trust they try to introduce a temporal dimension into traditional logic is needed. As a future work, they plan to develop combined logics of belief and time, on which trust theories can be based.

Marsh & Dibben (2005) claims that distrust is not a simple reversal of the concept of trust , although it is tightly coupled. It's also not *mistrust* or *untrust*, although again it's related. Mistrust, can be considered as either a former trust destroyed, or former trust healed. Untrust is a measure of how little the trustee is actually trusted. This is not quite the same as being the opposite of trust. Untrust is positive trust, but not enough to cooperate. Distrust is a measure of how much the truster believes that the trustee will actively work against them in a given situation. Thus, if I distrust you, I expect you'll work to make sure the worst . Distrust is a negative form of trust. If distrust is active, and allows the distruster to know that a trustee is not to be trusted in this situation. Distrust is a negative measure. In figure 2.5, the diagram serves to illustrate where our definitions of untrust, distrust and trust lie. Mistrust doesn't fit on

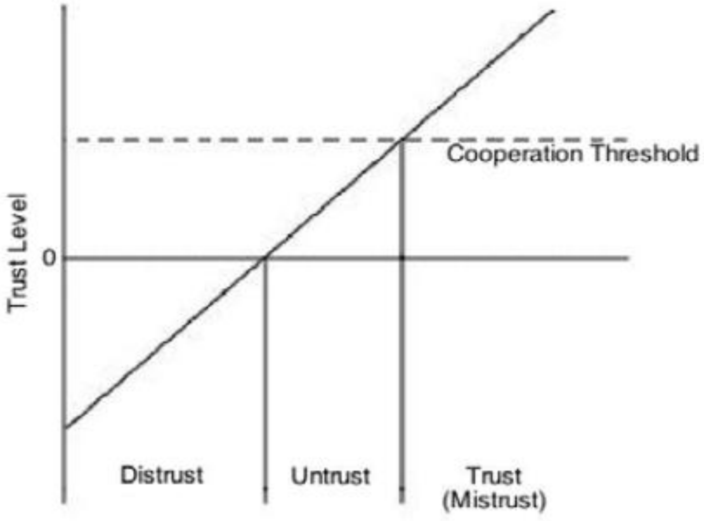


Figure 2.5 From distrust to trust, Marsh & Dibben (2005)

this diagram because it’s a misplaced value that was positive and misplaced.. Distrust really can be important in high risk situations, limiting exposure, being more risk averse, and exposing more gradually in risky situations than trust would result. Authors also claim that c*onfidence* is indicated by a lack of consideration for the risks involved . *Trus*t is indicated by a consideration of the risks involved.

Michalakopoulos & Fasli (2005) claim that under certain conditions, the trust dispositions are not important. Remembering past experiences for ever is not beneficial for the agents. In most cases optimism is good when the market consists mainly of reliable sellers. Pessimism is good when the majority of the agents are unreliable. In the case of risk neutral agents, making higher profits than the risk averse ones in an uncertain marketplace, can be explained by taking into account the fact that the agents do not make blind decisions about where to buy their goods. But they take into account both their trust towards sellers and their risk behaviour.

Wei-Peng & Ju (2008) propose formal definition of trust and security of task oriented information system. They assume that the trust has detailed information of prerequisites, behaviors and their relationship and the security be the implementation of target trusted behaviors based on the trusted relationship of system.They give a directed graph to describe the trusted relationship as shown in figure 2.6. With the formal model of trust and security, they can analyze a task-oriented information system formally. They define the trusted module and its interface, and describe a multi-layer trusted structure to help to avoid illegal trusted relationships.

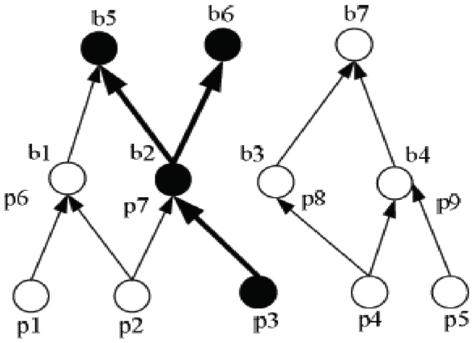


Figure 2.6 Directed graph of a simple system, Wei-Peng & Ju (2008)

Haque & Ahamed (2007) propose Hop Based Recommendation Protocol (HBRP) for distributed systems.This protocol includes mechanisms for active and passive recommendations. The format for a Hop Based Recommendation Request packet is as follows:

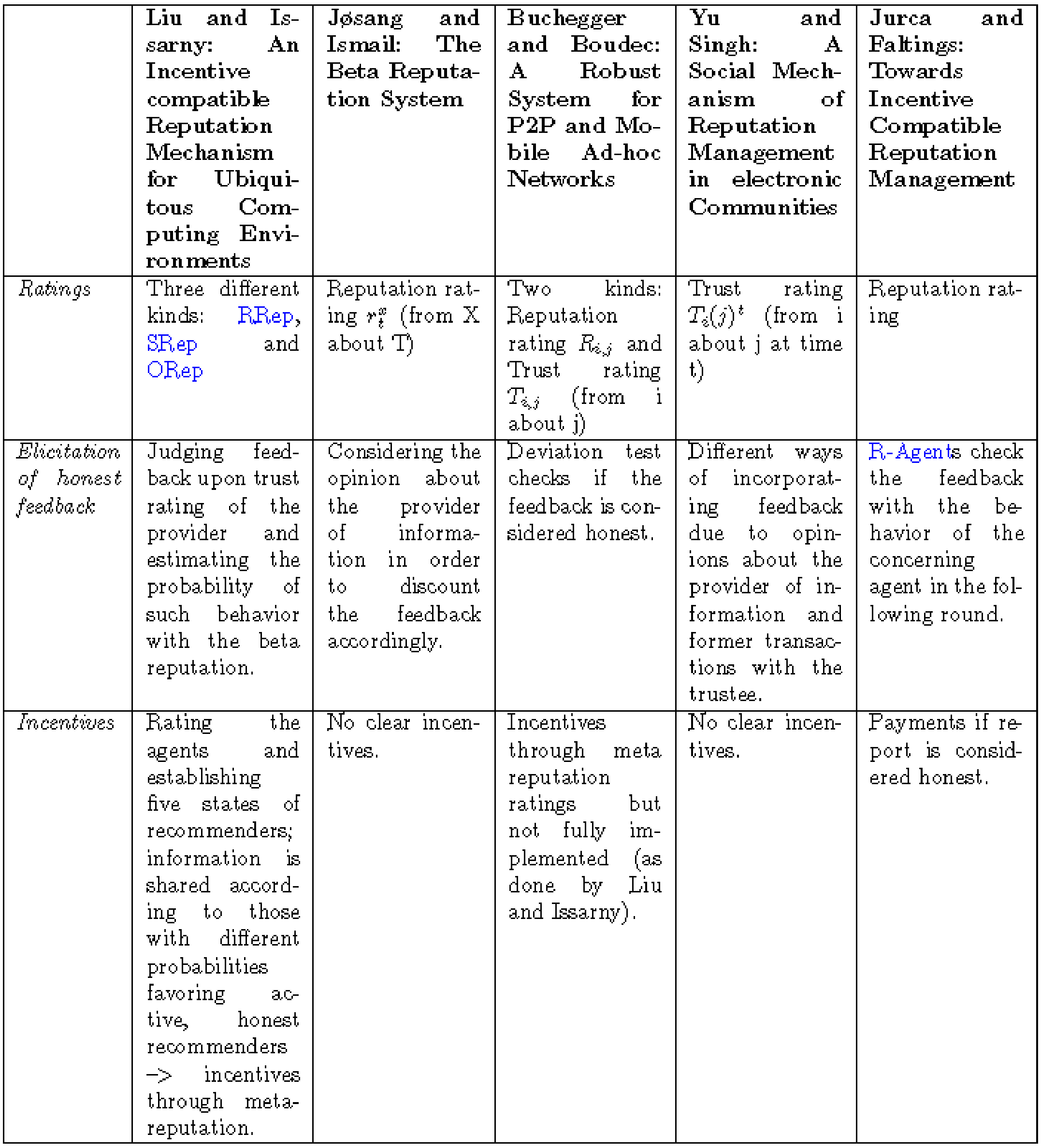
HBRReq=(Req_ID, SP_ID, SR_ID, IH, IR, TS). The hop field (IH) defines the maximum path length for the recommendation request This enables a node to avoid a long chain of recommendations. This value is reduced in each hop by 1 and the path is ignored when the field becomes 0. The IR field contains the trust value of the first link over the path. The TS field has been used to restrict a replay attack. The reply packet has the following format:

HBRRep=(Req_ID, Rec_ID, RH, TR, TS). Rec_ID denotes the node that is providing the reply to SP. The RH field shows the hop value which has been formed by reducing the IH value by one in each hop. The TR field sums up the trust value over the path.

Ray & Chakraborty (2009) propose a model that allows to formalize trust relationships. The trust relationship between a truster and a trustee is associated with a context and depends on the experience, knowledge, and recommendation that the truster has with respect to the trustee in the given context. They show that their model can measure trust and compare two trust relationships in a given context. Sometimes enough information is not available about a given context to evaluate trust. In this case, they show how the relationships between different contexts can be captured using a context graph. Formalizing the relationships between contexts allows to derive values from related contexts to approximate the trust of an entity even when all the information needed to calculate the trust is not available. They also show how the semantic mismatch that arises because of different sources using different context graphs can be resolved and the trust of information obtained from these different sources compared.

Heitz & König (2009) explain the resarch they realized about reputation assesment mechanisms. They summarize the results as given in table 2.3.

Table 2.3 Summary of reputation mechanisms, Heitz & König (2009)



	Liu and Issarny: An Incentive compatible Reputation Mechanism for Ubiquitous Computing Environments	**Jøsang and Ismail: The Beta Reputation System**	**Buchegger and Boudec: A Robust System for P2P and Mobile Ad-hoc Networks**	**Yu and Singh: A Social Mechanism of Reputation Management in electronic Communities**	**Jurca and Faltings: Towards Incentive Compatible Reputation Management**
Ratings	Three different kinds: RRep, SRep and ORep	Reputation rating r_t^x (from X about T)	Two kinds: Reputation rating $R_{i,j}$ and Trust rating $T_{i,j}$ (from i about j)	Trust rating $T_i(j)^t$ (from i about j at time t)	Reputation rating
Elicitation of honest feedback	Judging feedback upon trust rating of the provider and estimating the probability of such behavior with the beta reputation.	Considering the opinion about the provider of information in order to discount the feedback accordingly.	Deviation test checks if the feedback is considered honest.	Different ways of incorporating feedback due to opinions about the provider of information and former transactions with the trustee.	R-Agents check the feedback with the behavior of the concerning agent in the following round.
Incentives	Rating the agents and establishing five states of recommenders; information is shared according to those with different probabilities favoring active, honest recommenders -> incentives through meta-reputation.	No clear incentives.	Incentives through meta reputation ratings but not fully implemented (as done by Liu and Issarny).	No clear incentives.	Payments if report is considered honest.

In the table, transitivity value indicates whether this trust can be passed on to a third party or not. In this model, trust can only be transitive or intransitive in a specific context.

Ajayi, Sinnott, & Stell (2007) propose Dynamic Trust Negotiation (DTN) framework. DTN is the process of realising trust between strangers or two non-trusting entities, e.g. institutions, through locally trusted intermediary entities. Trust is realised when an entity delegates its digital credentials to trusted intermediary entities through which it can interact with non-trusted entities. This intermediary entities can in turn delegate to other intermediary entities resulting in what we call n-tier delegation hops. The trust negotiation process involves trust delegations through intermediary trusted entities on behalf of non-trusting entities. Any entity can serve as a negotiator for other entities provided it is trusted by the two non-trusting entities or by their intermediaries. DTN negotiates credentials between trusted parties also known as a circle of trust COT, who act as mediators on behalf of strangers and thus bridge trust gaps. This bridge also reduces the risk associated with disclosing policies to strangers. Cicle of trust example is shown in figure 2.7.

.

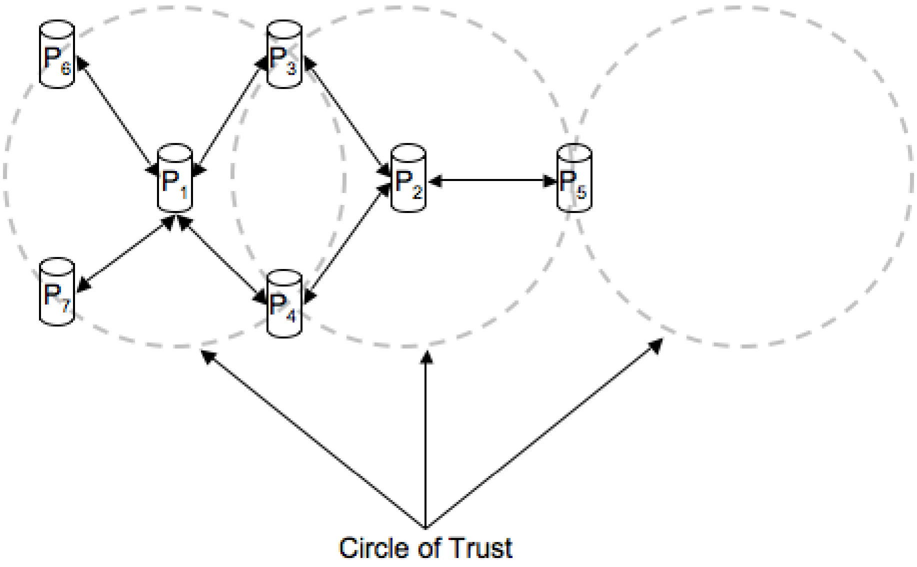
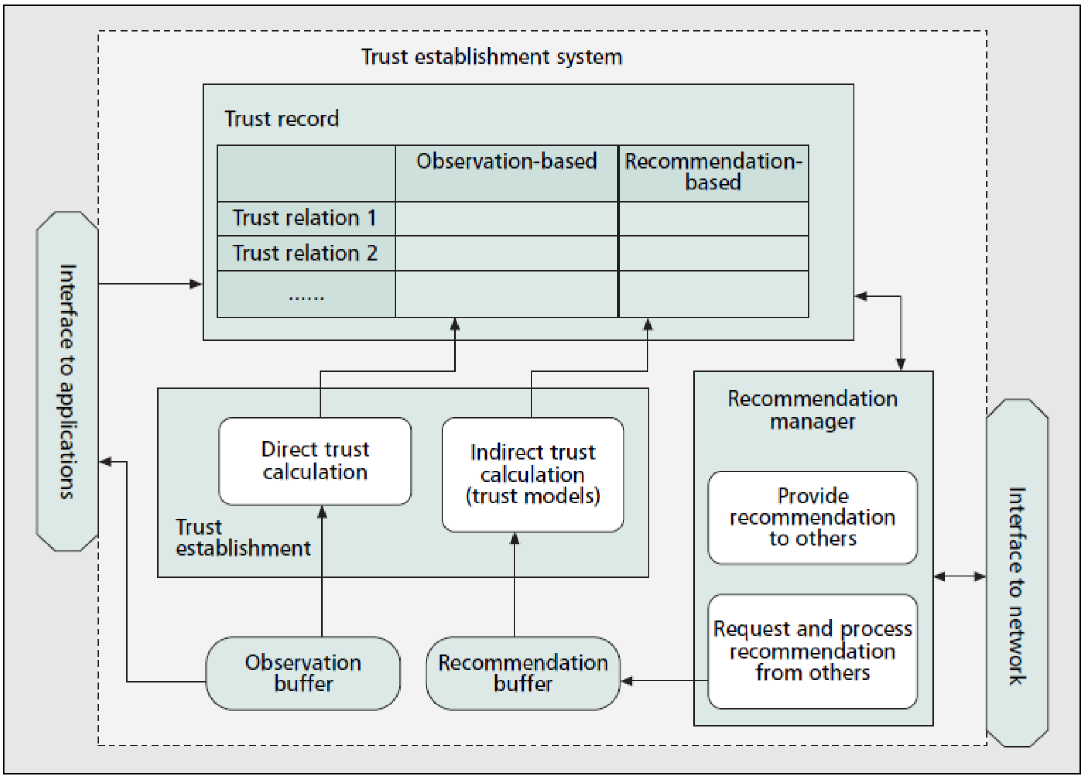


Figure 2.7 Circle of trust, Ajayi, Sinnott, & Stell (2007)

In dynamic trust negotiation (DTN), credentials are only disclosed to intermediary parties, which are trusted with the expectation that privileges would be delegated to it that wouldn’t be directly to non-trusted parties. Further as negotiations take place from one intermediary party to another, the privacy of the requester is even more protected.

Sun, Han, & Liu (2008) proposes for MANETs and sensor Networks a distributed trust managementmodel, where each network entity maintains a trust manager. The basic elements of such a trust manager are illustrated in figure 2.8. *The trust record* stores information about trust relationships and associated trust values. A *trust relationship* is always established between two parties for a specific action. For each trust relationship, one or multiple numerical values, referred to as *trust values*, describe the level of trustworthiness. *Direct trust* is established through observations.


Trust establishment system
Trust record
Interface to applications
Trust establishment
Direct trust calculation
Indirect trust calculation (trust models)
Observation buffer
Recommendation buffer
Recommendation manager
Provide recommendation to others
Request and process recommendation from others
Interface to network

	Observation-based	Recommendation-based
Trust relation 1		
Trust relation 2		
......		

Figure 2.8 Basic elements in trust establishment systems, Sun, Han, & Liu (2008)

The previous interactions between the subject and the agent are successful and *Indirect trust* is established through trust propagation. Two key factors determine indirect trust. The first is when and from whom the subject can collect recommendations. The second is to determine how to calculate indirect trust values based on recommendations. Malicious parties can provide dishonest recommendations to frame good parties and/or boost trust values of malicious peers. This attack, referred to as the *bad mouthing* attack, is the most straightforward attack.

On-off attack means that malicious entities behave well and badly alternatively, hoping that they can remain undetected while causing damage. This attack exploits the dynamic properties of trust through time domain inconsistency.

Canfora, Costante, Pennino & Visaggio (2008) propose an approach aims at utilizing a front-end trusted filter, which allows the access to data only when the data privacy policy is not violated. In order to apply the approach,authors developed a prototypal system named *DataGateKeeper*. The system acts like a Proxy between the data requestors and the data providers. Data requestors could be humans, devices or other software systems, which seek for information and send queries to the data providers. Instead of sending the query directly to the data providers, the data requestors send the query to the DataGateKeeper. This solution is transparent to the data requestor and it does not involve the organization or the presentation of data. Proposed model is shown in figure 2.9.

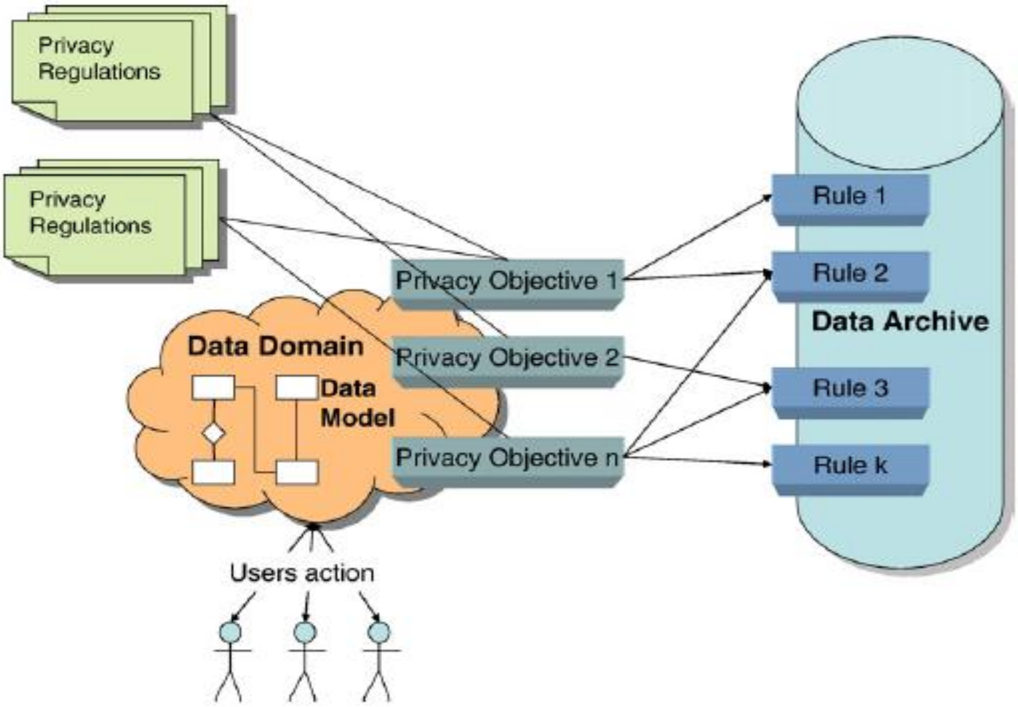


Figure 2.9 The proposed model, Canfora, Costante, Pennino & Visaggio (2008)

Barber, Fullam, & Kim (2002) sought to challenge the community working on issues of trust, fraud and deception in the Multi-Agent Systems. In performing trust model component discrimination, a unified set of trust elements must be defined. Furthermore, algorithms must be developed for distinguishing between these elements as behavior causes. Trust component discrimination can then be utilized for the development of prioritization strategies, in which agents can choose with whom to interact based on the components on which it places importance.

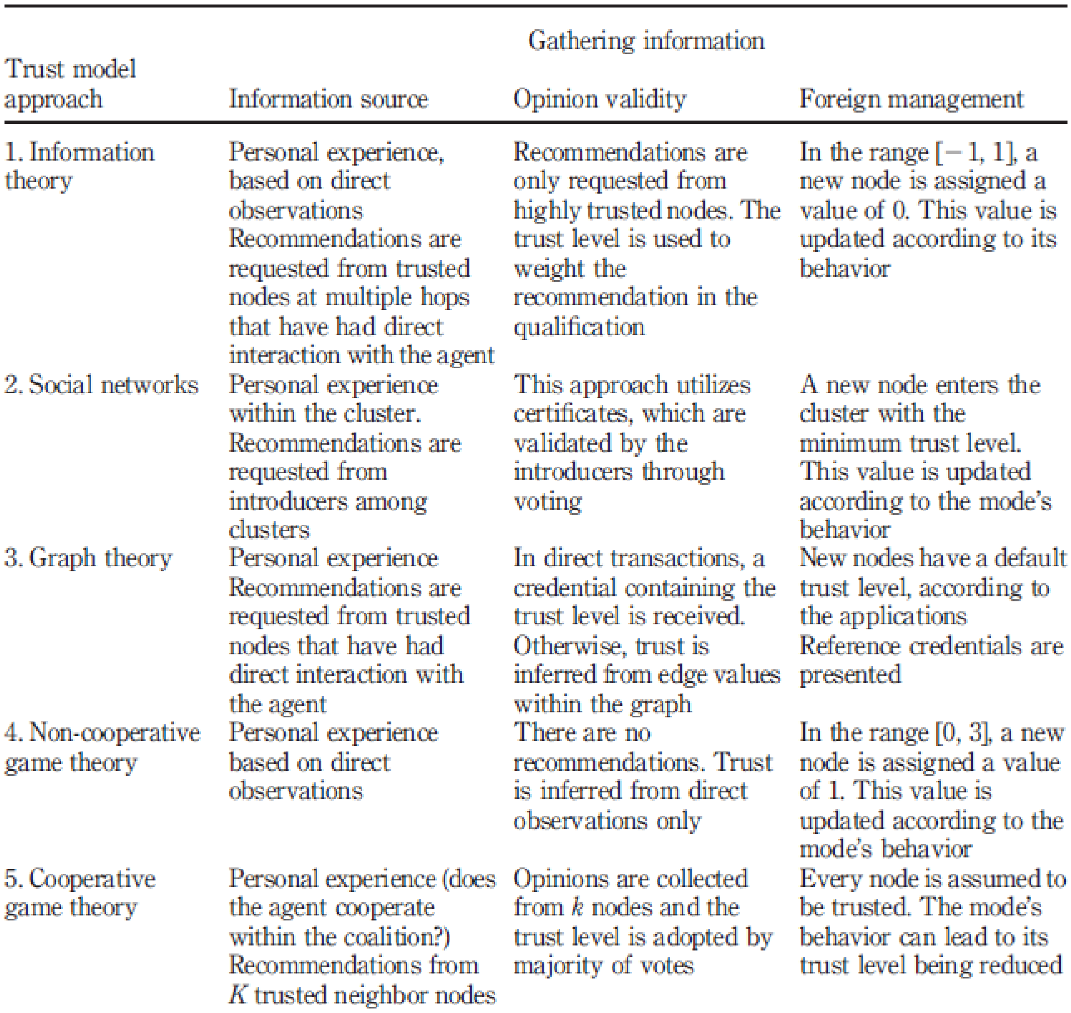
- In developing alternatives to interaction-based reputation building, low-risk, noninteraction based strategies must be enumerated and defined, then integrated and evaluated.
- Examination of human factors in reputation modeling, including prejudice, gossip, and first impressions, can provide a launch point for strategy development, as well as insight into possible strategy flaws.
- Benchmarking trust first requires a defined set of metrics and a normalization of trust representations. Then, existing algorithms can be evaluated against those benchmarks to assess the feasibility of building upon them. Algorithm performance evaluation shows the way for choosing trust strategies to custom fit parameters, through dynamic trust maintenance strategy selection.

Weeks (2001) presents a mathematical framework for expressing trust management systems. The framework in his work can be used to explain existing trust management systems and to help design new ones. It can provide a precise specification of the semantics of a trust management system, which is important for building correct, interoperable implementations. The least fixpoint semantics leads to implementations of trust management engines. The framework can concisely specify trust management systems by an authorization lattice and language for licenses. This makes it possible to compare the expressiveness of systems. It also makes it easier to assess the applicability of a system to a given situation and to analyze design tradeoffs among current and new systems. The framework can also help to improve existing languages for expressing licenses by making them more regular and more expressive.

Budalakoti, DeAngelis, & Barber (2009) propose a recommender for selecting the most appropriate responders given a question. This recommender is the core of a question and answer forum under development that is designed to encourage expert participation. The two primary contributions of this work are a finite mixture model based approach for characterizing the production of content in an online question and answer forum and, a decision theoretic framework for recommending expert participants while maintaining questioner satisfaction and distributing responder load. Their generative model uses word content information and collaborative information to build models of users expertise, which are employed during recommendation. They have also developed two new metrics: *responder load* and *questioner satisfaction*.

Mejia, Pena, Munoz, & Esparza (2009) are focused specifically on trust models for promoting cooperation in ad hoc networks and it analyzes the most recent research in this area. A comparative analysis of the trust models, emphasizing the methods utilized by each model to carry out the three tasks described in table 2.4 . It shows a comparison of the data collection task for each trust model. Each approach uses experience as the main data source, taking advantage of a characteristic of wireless networks, whereby all nodes can listen to the information transmitted within their reception range. However, four of these approaches also use references of neighbor nodes as an additional data source and, although each approach has a particular way of collecting and validating recommendations, the purpose is the same in every model. With respect to the treatment given to new nodes entering the network, all approaches show a common factor in their policies. All of them determine some minimum trust level assigned to the new node and allow this value to change according to behavior. Thus, the new node can become part of the network, creating its own history record and collaborating with the distributed functions, or it can be isolated by its malicious or selfish behavior.

Table 2.4 Methods of gathering information at ad-hoc networks, Mejia, Pena, Munoz, & Esparza (2009)



Trust model approach	Gathering information Information source	 Opinion validity	 Foreign management
1. Information theory	Personal experience, based on direct observations Recommendations are requested from trusted nodes at multiple hops that have had direct interaction with the agent	Recommendations are only requested from highly trusted nodes. The trust level is used to weight the recommendation in the qualification	In the range [− 1, 1], a new node is assigned a value of 0. This value is updated according to its behavior
2. Social networks	Personal experience within the cluster. Recommendations are requested from introducers among clusters	This approach utilizes certificates, which are validated by the introducers through voting	A new node enters the cluster with the minimum trust level. This value is updated according to the mode's behavior
3. Graph theory	Personal experience Recommendations are requested from trusted nodes that have had direct interaction with the agent	In direct transactions, a credential containing the trust level is received. Otherwise, trust is inferred from edge values within the graph	New nodes have a default trust level, according to the applications Reference credentials are presented
4. Non-cooperative game theory	Personal experience based on direct observations	There are no recommendations. Trust is inferred from direct observations only	In the range [0, 3], a new node is assigned a value of 1. This value is updated according to the mode's behavior
5. Cooperative game theory	Personal experience (does the agent cooperate within the coalition?) Recommendations from K trusted neighbor nodes	Opinions are collected from k nodes and the trust level is adopted by majority of votes	Every node is assumed to be trusted. The mode's behavior can lead to its trust level being reduced

Raya, Papadimitratos, Gligor, & Hubaux (2008) propose a framework for data-centric trust establishment. Trust in each individual piece of data is computed then multiple related but possibly contradictory data are combined. Finally, their validity is inferred by a decision component based on one of several evidence evaluation techniques.Authors consider and evaluate an instantiation of framework in vehicular networks as a case study. Simulation results show that framework is highly resilient to attackers and converges stably to the correct decision. Flowchart of data-centric trust establishment framework is shown in figure 2.10.

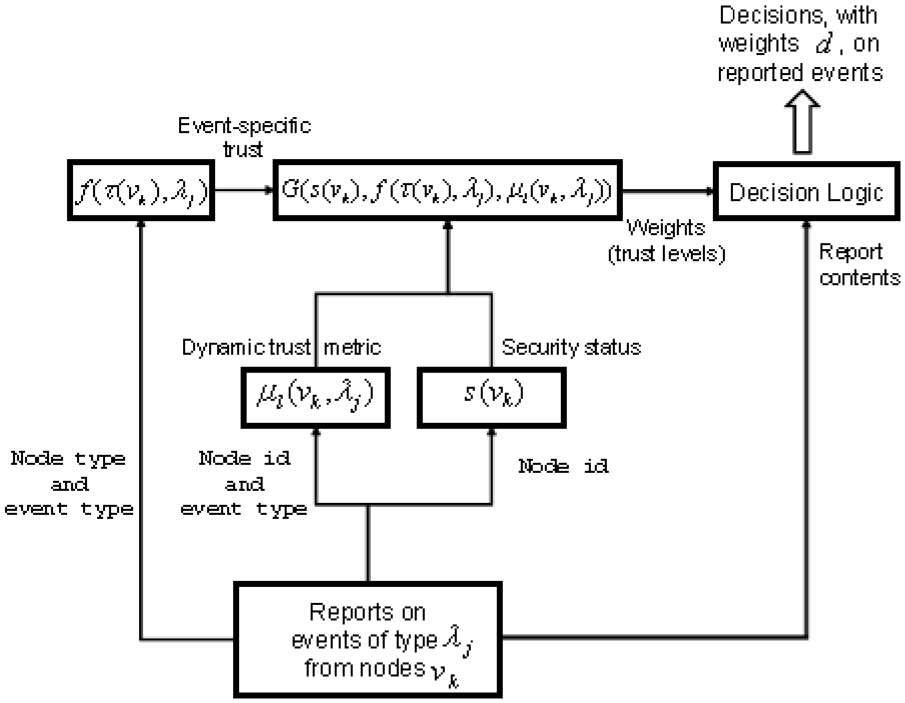


Figure 2.10 Data-centric trust establishment framework Raya, Papadimitratos, Gligor, & Hubaux (2008)

Thiagarajan, Raghunathan, Natarajan, Poonkuzhali, & Ranjan (2009) propose a trust rating system for distributed networks. Distributed network is considered as a signed graph. Each node in the graph is considered as an agent and each edge is assigned with a weight called *precedence of acceptance*. Signed weight (++, +-, -+, --) is attached to each node of the graph.Value attached based on the agent-client combination. The sign assigned to the agents tend to change according to the client with which it interacts. Initially, all the agents are assigned with ++ weight. Then the client is allowed to give their precedence of acceptance (++, +-,-+,--) over the specified agent. Depending upon the precedence given by clients, the trust level rating of agents having positive trust with positive attitude is calculated. Similarly agents having positive trust with a negative attitude, negative trust with positive attitude and negative trust with a negative attitude is estimated. Similarly, all the clients are assigned with ++ weight initially. Then the agent is allowed to give their precedence of acceptance (++, +-,-+,--) over the specified client. Depending on the precedence given by agents, the trust level rating of clients having positive trust with positive attitude is calculated. Similarly clients having positive trust with a negative attitude, negative trust with positive attitude and negative trust with a negative attitude is estimated.

Yolum & Singh (2004) propose a trust model for large-scale, decentralized information systems that are represented by autonomous agents. They group trust establishment methods in three major groups:

- Institutional trust.
- Social trust (based on local or social evidence).
- Trust based on referrals.

They propose two graph types for representing their model:

- A vector space model. Each element in the vector corresponds to a different domain and the weight of the element denotes the trustworthiness of the service for that domain
- A service graph model. A service graph is maintained by each agent to autonomously capture its experiences. Thus agents may have differing weights for the same pair of services. The weights are adjusted independently by each agent. A simple service graph is shown in figure 2.11. Some experimental results based on the service-graph model are given in the paper.

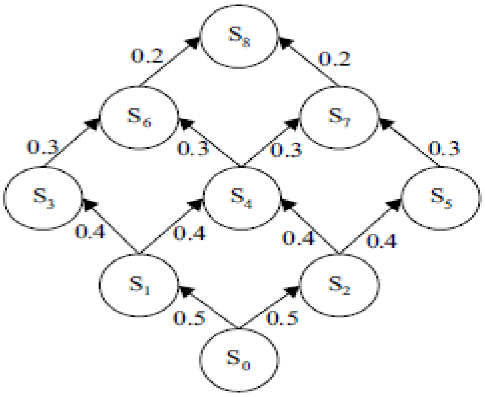


Figure 2.11 A simple service graph, Yolum & Singh (2004)

Wang & Singh (2007) propose a formal representation of trust for distributed multiagent networks. Their work mainly depends on Josang's previous work on *probability certainty* distribution. But they change the definition of two operators, *concatenation* and *aggregation*. They give mathematical properties of these operators and prove them.These properties are:

- The concatenation operator $\otimes$ is associative.
- The aggregation operator $\oplus$ is associative.
- The aggregation operator $\oplus$ is commutative.
- The concatenation operator $\otimes$ does not distribute over the aggregation operator $\oplus$.

By using these two operators, they calculate the trust rating as a path algebra Problem.The direction of calculation is from bottom to top.

Hang, Wang, & Singh (2008) propose a frame work based on the work in Wang & Singh (2007). In addition to aggregation $\oplus$ and concatenation $\otimes$ operators they define a new operator selection Ⓢ. Some properties of selection operator are:

- Selection operator Ⓢ is commutative.
- Selection operator Ⓢ is associative.
- Concatenation operator $\otimes$ distributes over selection operator Ⓢ.

The aim of the selection operator Ⓢ is to select one out of multiple paths that end at the same point. The path that gives the highest belief is selected. So that the problem of double-counting is prevented. By using the only aggregation operator on multiple paths from the same witness can lead to double-counting. The aggregated belief of the paths from the source to a witness may be greater than one when double-counting occurs.

Hang & Singh (2008) investigate the problem of selecting services based on criteria such as user requirements and service qualities. They define *trust-aware service selection* for selecting desired services. Selection is based on the trust placed in their ability to deliver specified values of the specified qualities. A trust-aware service selection should support the following criteria:

- Selecting service instances should be based on the qualities desired.
- Selected services should be rewarded and punished in an appropriate manner. So that the best information needed to support successful compositions could be maintened. Trust-aware service selection framework is shown in figure 2.12.

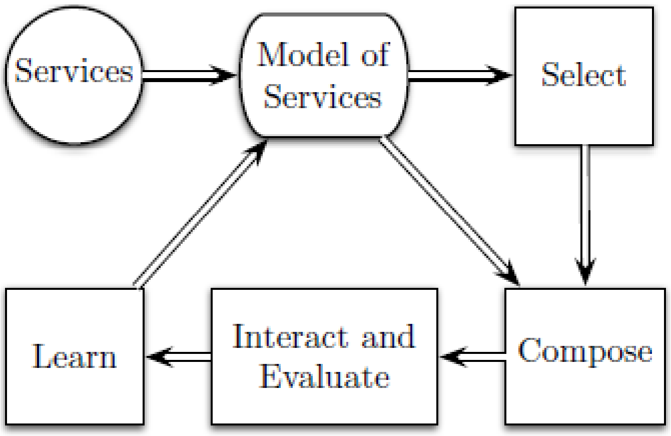


Figure 2.12 Trust-aware service selection framework, Hang & Singh (2008)

They use two different computational methods for their framework:

- *Bayesian approach:* It models service compositions by using Bayesian networks in partially observable settings. Bayesian approach captures the dependency of providing good service between composite and underlying services. It also adaptively updates trust to reflect most recent quality.
- *Beta-Mixture approach:* This approach can learn the distribution of composite quality and also the underlying services' responsibility in composite quality without actually observing the underlying performance.

These two approaches provide different information about services. Bayesian approach uses online learning to track the service behavior. It also tells consumers how good service they can expect from composition when the underlying services are good. Beta-mixture model learns the quality distribution of services and provides how much each underlying service contributes in the composition.

Christopher & Singh (2010) propose a model to asses the trustworthiness of other agents. Because, today in e-commerce transactions are automated and the risk being cheated increases.They claim that agents with high measured discount factors often behave in a trustworthy manner.They offer a mathematical model that discount factors is a measure of trustworthiness .

Holtmanns & Yan (2006) analyse social trust scenarios and try to derive abstract trust concepts. From these abstract trust concepts a context-aware adaptive trust concept is developed. The context-aware adaptive trust concept takes into account the dynamics of trust and the context based grouping of trust properties. The adaptive trust graph can be grouped into context based sub-graphs based on the non-zero rights that are connected to the different resources. A group of resources build the actual user context. An example of adaptive trust graph is give in figure 2.13.

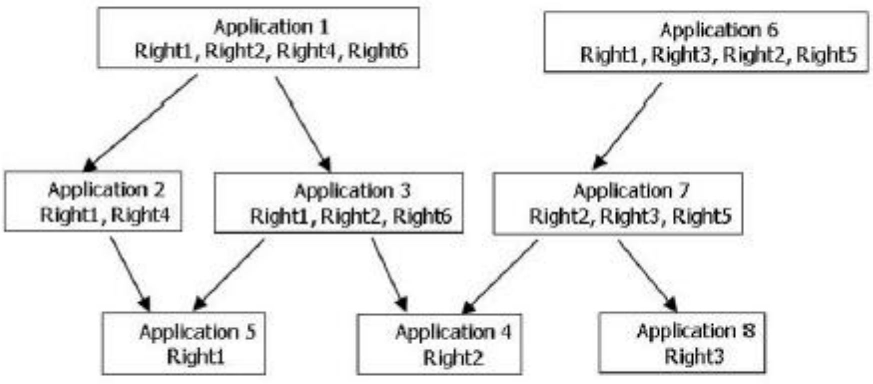


Figure 2.13 A simple adaptive trust graph, Holtmanns & Yan (2006)

Neisse, Wegdam, & Sinderen (2006) propose context-aware trust domains as a management solution for context-aware service platforms. This work is done as a part of a prototype implementation in the AWARENESS project. They divide trust in three different aspects as the social,the informational, and the technical They are mainly focused on informational aspects as shown in figure 2.14. The main purpose of their work is to reduce the complexity in the management of trust relationships using the abstraction of context-aware domains. So that, trust degrees do not have to be specified individually for each entity, but in a set for a collection of entities part of a context-domain.

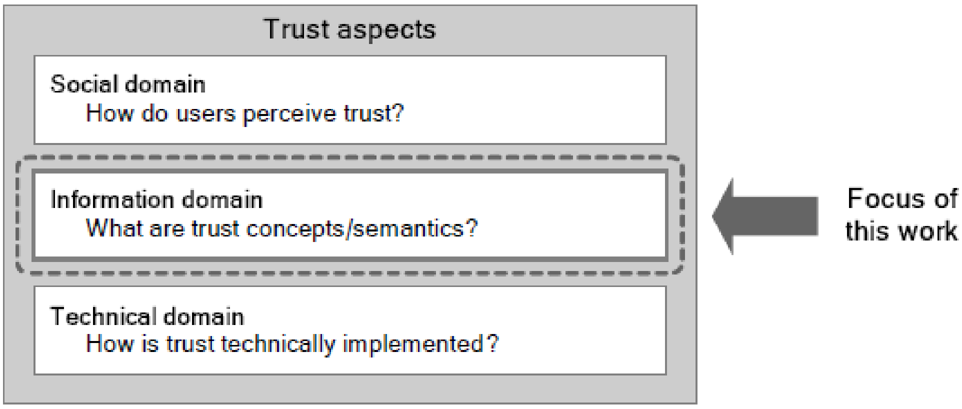

Trust aspects
Social domain
How do users perceive trust?
Information domain
What are trust concepts/semantics?
Technical domain
How is trust technically implemented?
Focus of this work

Figure 2.14 Trust aspects, Neisse, Wegdam, & Sinderen (2006)

Zia (2008) proposes a framework which uses the reputation and trust management to detect trust behaviour, on the basis of the responses from other neighbouring nodes in wireless-sensor networks. If the number of trust entries concerning a particular node reaches a set threshold, that node is declared un-trust. This message is broadcasted, alarming all the neighbours and eventually reaching the base station. The cluster leader or base station then isolates the un-trust node and all traffic coming from that node is ignored. He also made a simulation to measure the response time.The simulation results show that the time it takes to detect a untrust node is decreased when there are more nodes in the network.

Andersen and et al. (2008) analyses networks dealing with high quality personalized recommendations. These systems often have explicit knowledge of social network structures that represent trust and recommendation. The goal of trust-based recommendation systems is to generate personalized recommendations by aggregating the opinions of other users in the trust network. They compare the various algorithms used in such networks. Algorithms are give as follows:

- Random Walk System(RW).
- Majority-of-Majorities (MoM).
- Minimum Cut System (min-cut).

Yang and et al. (2002) proposes a W3 Trust Model (W3TM) to measure the trustworthiness of online services through evaluating the trust and transitivity of trust of Web contents. The W3 Trust Model brings the concepts of trust and transitivity of trust into an analysis of front-end Web contents using a proposed trust evaluation process. Targeted site is based on the result of recursive calculation of the following component assessments:

- Standalone page trust assessment.
- Relevance assessment among hyperlinked pages.
- Subordinate node assessment.

Chen and et al. 2005 proposes a trust model for multi-agent system using fuzzy sets (TMMASFS). There are three kinds of trusts in TMMASFS: the direct trust, the recommendation trust and the self-recommendation trust. TMMASFS overcomes the shortcomings of the trust models, and is adapted to the uncertain network environment more effectively. The distiguishing feature of TMMASFS is the self-recommendation trust. The self-recommendation trust is useful very much when the manager agent has no direct experience or recommendation about the contractor agents. This model is efficient and adapted to the dynamic and uncertain network enviroment.

CHAPTER THREE
GRAPH BASED TRUST MODEL

3.1 Motivation

Our research work aims to build *a Trust Modelling for an Environment in which subjects would like to asses their trusts on objects.* It's main component will be *Trust Graphs. Trust graphs* can be described as the *Graphical Modeling of Trust Relationships*.

Knowing how much to trust someone helps us know what to do in our interactions with them. The main motivation for this work is to keep security and privacy of users' networks. User networks are in continous interaction with the following entities.

- *Global internet:* Today's internet is an example of an open global network. Communication occurs across various boundaries (Madigan and et al., 1997). These are topological, organisational, political and geographical boundaries. Users in the communication may not be known before and may never meet physically in the future.

- *Ubiquitous connectivity*: The mobile and wireless technology connnects people and also the objects (devices) all over the world (Nam, 2009). This allows that some applications can controll objects remotely via the network. Groups of objects can collaborate in an ad-hoc manner for various tasks. Intermittent connectivity and short-lived relationships are characteristic of such systems.

- *Software agents*: Increasingly, tasks are delegated to software. Examples can be given such as automated notification of news items, purchasing items online and matching user preferences etc (Esfendiari and et al., 2001). The

- agent must be sure that the other agents it communicates are trustworthy enough.

- *Assesing trust mathematically to web services:* Assesing trust mathematically to web services give us a numerical value. Numerical values can be used in the rapid assestment of the trustworthiness of the web services (Wang and et al., 2010). This process is in the behalf of the user because malicious web services can be detected immediately and the user is prevented from undesired deceptions.

Trust plays a central role in the security of interactions in the systems described above. The framework presented in this research will provide the *formalization of trust as a computational concept.*

3.1.1 Trust Definition of Us

The term trust has a very general meaning. At the first step, we must clarify the limitations of our model. Otherwise confusions about trust relationships may arise.

We define *trust* as following:

Definition 3.1 : Trust is the expectation of an entity from another entity based on a predifined set of contexts in the specified time interval.

3.2 Trust Model as an Entity-to-Entity Graph

When one says that *'An Entity A trusts another entity B within a context C'*, a formal representation of trust involving A, B, C need to be given. Formal representation that is chosen in this thesis is an entity-to-entity graph. Entity-to-entity graphs will later be converted into subject-to-object graphs, bipartite graphs and colored graphs. As a first step, we consider the simplest case. A *Trust Graph* is a labelled graph in the of the form $G=(V(G),E(G))$, where $V(G)$ represents the entities and $E(G)$ represents trust assesments as edges of the graph.

Each edge $e=(v_i,v_j)$ in $E(G)$, $E(G)\subseteq V(G)xV(G)$ means the entity v_i has a trust relationship with the the object v_j and has a edge label (l). l is the feature or the object on which the trust assesment is made. In the simplest case edge label l is the context on which trust assesment is made. The simplest case for *Entity-to-Entity Graph* is shown in figure 3.1.

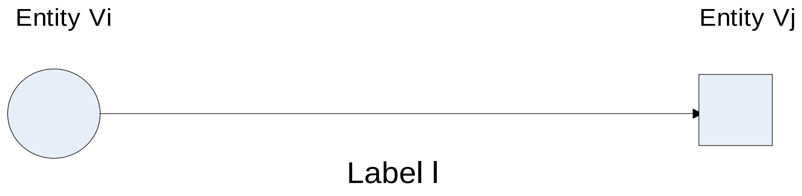


Figure 3.1 An entity-to-entity graph in its simplest form

Definition 3.2 (Entity-to-Entity Graph): An entity-to-entity graph is in the form of $G=(V(G),E(G))$ where a non-empty set of graph vertices V is :

$V=\{v_1,v_2,v_3,...,v_n\}$; n is the number of vertices and a non-emty set of graph edges is:

$E\subseteq VxV$ where $e=(v_i,v_j)\in E$ represents an edge from vertex v_i to vertex v_j. An edge could be directed or not directed. Also it could be labelled or not labelled. Entities could be subjects or objects. Entity-to-Entity graphs can be in the form of:

- Subject-to-object
- Subject-to-subject
- Object-to-object

On the basis of above premises, the following trust forming factors can be identified.

- *Entity:* An entity is a unit which is aware of other entity's trustworthiness. It also has the ability to decide under which conditions to set up interactions with other entities. An entity can be a person, an agent, a host, a device, a process, a service etc.

- *Trust*: Our trust definition is given in definition 3.1.

- *Edge Label* l: Edge label defines the trust attributes on which trust assesment is made. In our model a label has three attributes: context *c*, trust metric *p* and the time specification *t.* Time *t* represents a time interval $t = \left[t_1, t_2 \right]$ on which trust assesments are made.

- *Context:* Trust is always based on a context. Annid K. Dey (2000), defines the context as *"any information that can be used to characterise the situation of entities. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves".* Contexts are divided into *direct* or *recommended* contexts to reflect the nature of the trustee in the relationship. Context is sometimes called as *trust scope*.

- *Trust Relationship*: A trust relationship can only exist between two entities. It reflects the truster's opinion about the trustee's trustworthiness. A trust relationship is uni-directional. If entity A trusts entity B and entity B trusts entity A, each trust relationship will be considered separately. A trust relationship is dynamic and may change over time.

Entity V_i trusts entity V_j within a context *c* with a trust metric *p*. *Trust Metric p* indicates the strength of the trust relationship between the entities V_i and V_j.

Definition 3.3 (Trust Metric): Trust metric *p* is a real number in the interval [0,1]. That means $p \in [0,1]$.

Entity-to-Entity Graph with a label (c,p) is shown in figure 3.2.

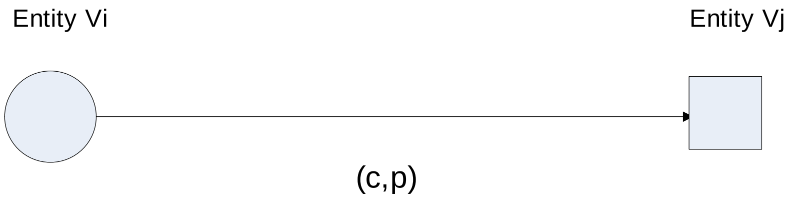


Figure 3.2 An entity-to-entity graph within a label (c,p)

Since trust relationship is dynamic and it may change over time, trust relationship must be defined within *a time specification*. In this thesis, time specification *t* is a time interval $t = [t_1, t_2]$. *Entity-to-Entity Graph* with a label (c,p,t) is shown in figure 3.3.

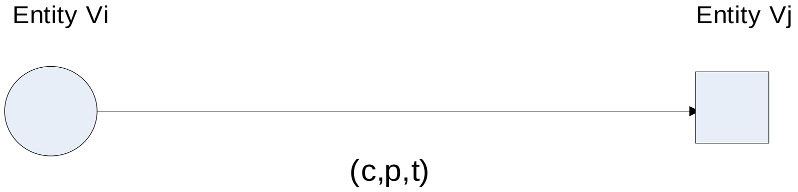
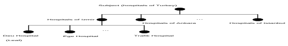
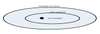


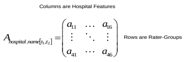
Figure 3.3 An entity-to-entity graph within a label (c,p,t)

3.3 Trust Model as a Bipatite Graph

In this section we will try to estimate trust relationships for helping decision of raters. Bipartite graphs will be used for modeling. A bipartite graph is a graph where nodes can be divided into two seperate groups U and V such that no edge connects the vertices in the same group.

Definition 3.4 (Bipatite Graph): A bipartite graph is composed of two non-empty distinct sets of U and V where $U = \{u_1, u_2, u_3, ..., u_n\}$; n is the number of elements of U and $V = \{v_1, v_2, v_3, ..., v_m\}$; m is the number of elements of V. A bipartite graph is shown as 

. A non-emty set of graph edges is: 

where 

Columns are Hospital Features
$A_{hospital.name[t_1,t_2]} = \begin{pmatrix} a_{11} & \cdots & a_{16} \\ \vdots & \ddots & \vdots \\ a_{41} & \cdots & a_{46} \end{pmatrix}$
Rows are Rater-Groups

represents an edge from vertex $u_i \in U$ to vertex $v_j \in V$.

Definition 3.5 (Subject to Object Graph): A subject to object trust graph $G = (U(G), V(G), E(G))$is a bipartite graph, which consists of a non-empty set of vertices $u_i \in U = \{u_1, u_2, u_3, ..., u_n\}$, $v_j \in V = \{v_1, v_2, v_3, ..., v_m\}$ such that U and V are the sets of subjects and objects respectively, and a set of directed edges $E \subseteq UxV$. Each directed edge $e = (u_i, v_j) \in E$ is labeled with a 3-tuple (c,p,t), where c is the trust context, p is the trust metric and t is the time specification

A Subject can be:

- a user
- an organization
- a host
- a device
- an agent
- a service
- a process

An object can be:

- an organization
- a host
- a device
- an agent
- a service
- a process

In our model we have two sets of entities:

- U : Subjects who rates the set V . Subjects set is composed of *n* subsets:
- V : Object which is rated by subsets of U . Subsets of the set V are the contexts which are rated by subject-subsets. V is composed of *m* subsets.

Our basic model has a number of subject-subsets related by edges to a number of contexts for a single object. Edges are always from a subject-subset to a context of the object. Each edge is a rating of a subject-subset for a context. In figure 3.4, circles represent subject-subsets, and squares represent the contexts of the object.

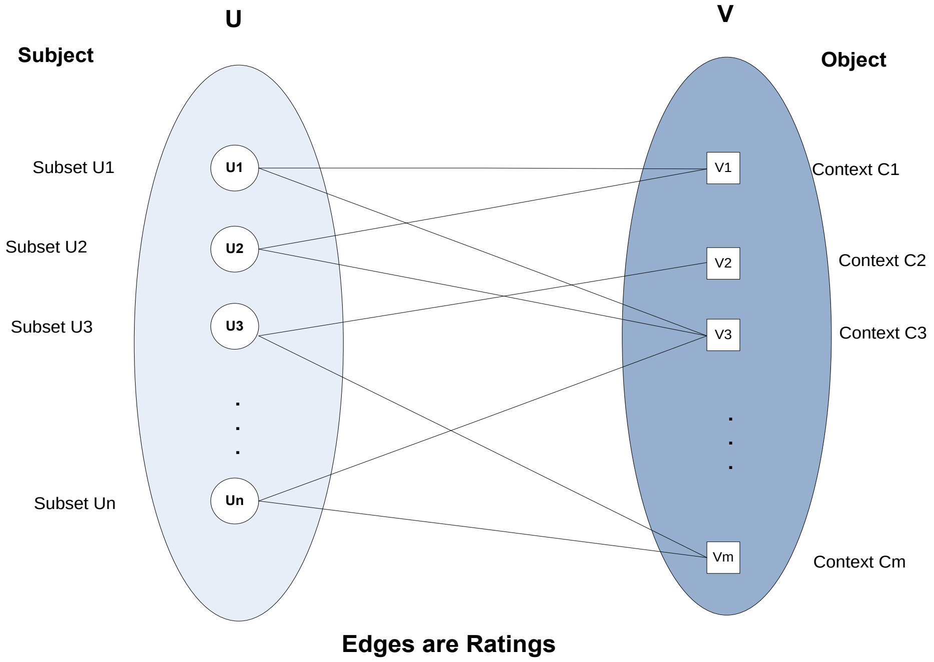


Figure 3.4 Bipartite-graph modeling

3.4 Hierarchical Structure of Subjects

Subjects generally are not a single entity and has many sub-sets. As the number of subjects involved in trust relationship increases complexity increases. Thus, our graphical model must be in hierarchical structure to decrease the complexity of interactions Hierarchical structure of subjects can be shown as in figure 3.5.

Subjects set $U = \{u_i \mid i = 1, n\}$ where u_i is either a member of the set U or a subset of U .

U is composed of union its subsets of U_i . $U = \{U_i \mid i = 1, k\}$ where $U_i \subseteq U$.

U can be shown alternatively as follows:
$U = \{U_{1-6}, U_{7-9}, U_{10-16}, U_{17-24}\}$.

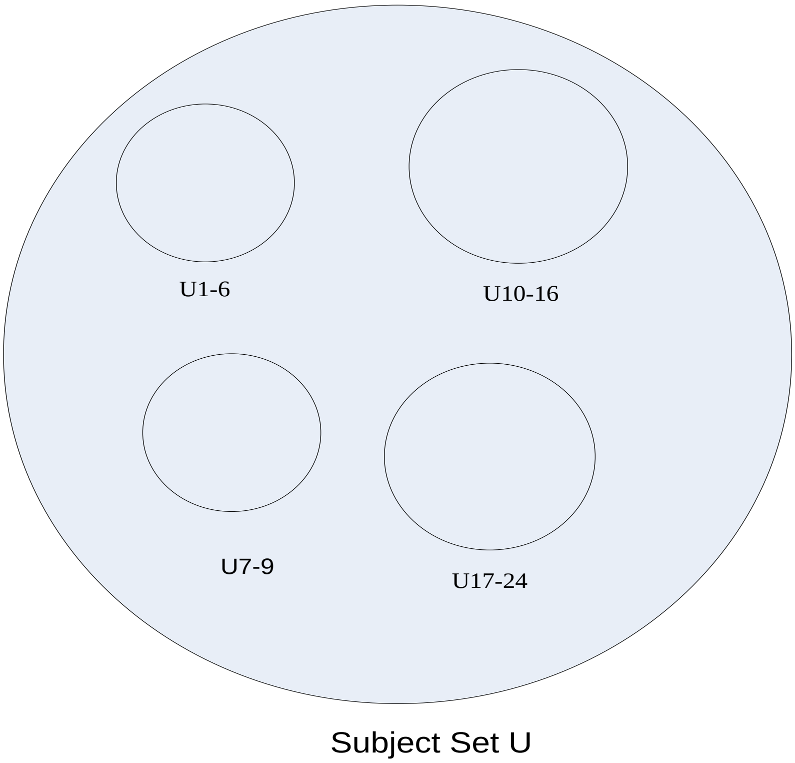


Figure 3.5 Hierarchical structure of subjects

3.5 Modeling Hierarchical Structure of Objects

We began describing the layout of the model as a bipartite graph of subjects and objects. Subjects set is represented by *U*. Our basic model has a number of subject-subsets represented by U_1, U_2, …, U_n. Subject-subsets are related by edges to a number of contexts for a single object set V. Object set V or represents *a single* entity and the subjects have access to a number of contexts for that object for rating. Contexts are the subsets of and V represented by V_1, V_2, …, V_m. Each context represent a feature of the object V. Obviously, subject-subsets do not rate for only a single object V. Object is the is only an element of a large group. So, we need an hierarchical model for classifying objects.

3.6 Tree-like Structure of Objects

In the hierarchical procedures, we construct a hierarchy or tree-like structure to see the relationship among objects. Root is the largest cluster containing all objects. Objects are classified in levels one-to-n under the root. Object level-n is at the bottom of the hierarchy and a leaf of the sub-cluster-n. Tree-like structure of objects is shown in figure 3.6.

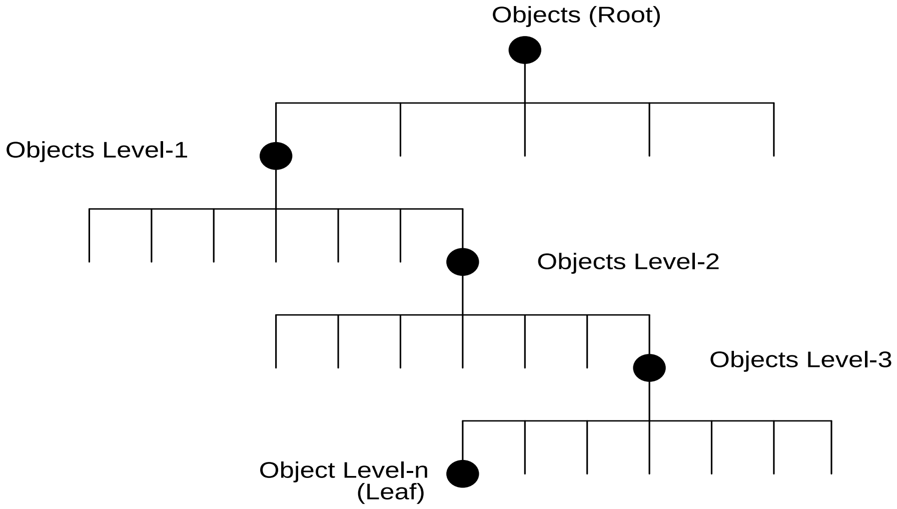


Figure 3.6 Tree-like structure of objects

Tree-like structure of the objects can be also represented as a Wenn-diagram as shown in figure 3.7.

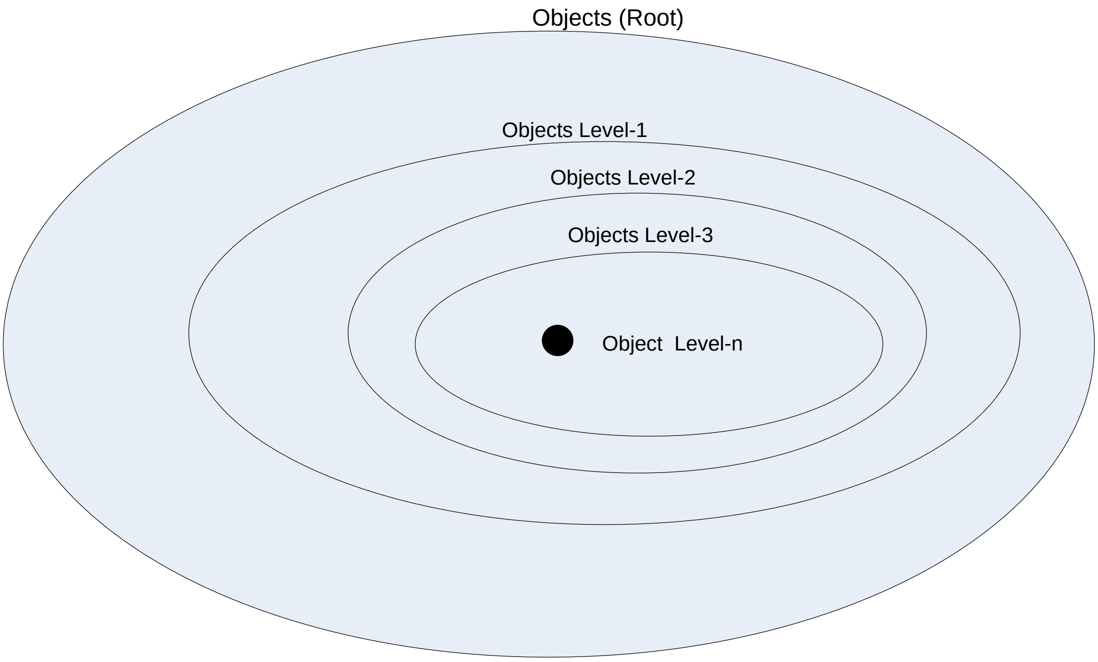


Figure 3.7 Wenn-diagram representation of the hierarchy of the objects

3.7 Construction of Assesment Matrix

Definition 3.6 (Assesment Matrix): Assesment matrix A_{nxm} defines the trust relationship of a bipartite graph at the time interval $[t_1, t_2]$ composed of two distinct sets of *U* and *V*, where *n* is the number of elements of the set *U* and *m* is the number of elements of the set *V*.

Assesment matrix allows us to compute the trust value of the subject *U* on the object *V*.

We can represent the trust relationship between the subject's set *U* and the rated contexts of the object V at the time interval $[t_1, t_2]$ as an *nxm* assesment matrix. Here n is the number of subsets of the subject's set U and m is the number of contexts of the object’s set *V*. Rows of the matrix represent *subject-subsets* and columns of the matrix represent *object* V ’s contexts as shown in figure 3.8.

$$\begin{array}{cc} & \text{Columns are Contexts of Object V} \\ A_{object.V[t_1,t_2]} = & \begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nm} \end{pmatrix} \text{Rows are Subsets of Subject U} \end{array}$$

Figure 3.8 Assesment matrix

Rows of the matrix represent:

$a_{11} - to - a_{1m}$: Subject-subset U_1

$a_{21} - to - a_{2m}$: Subject-subset U_2

$a_{31} - to - a_{3m}$: Subject-subset U_3

…

$a_{n1} - to - a_{nm}$: Subject-subset U_n

Columns of the matrix represent:

$a_{11} - to - a_{n1}$: Subject-subset U_1 ’s grade for the context V_1.

$a_{12} - to - a_{n2}$: Subject-subset U_2 ’s grade for the context V_2.

$a_{13} - to - a_{n3}$: Subject-subset U_3’s grade for the context V_3.

…

$a_{1m} - to - a_{nm}$: Subject-subset U_n ’s grade for the context V_m.

3.8 Coloring Trust Graphs

Assumption 3.1: Trust metric for the context V_m at the time interval $\left[t_1, t_2\right]$ increases as the number of subjects n rating the context increases.

Assumption 3.2: Each object V in the object's cluster has the same finite number of contexts $V_1, V_2, V_3, ..., V_m$ at the time interval $\left[t_1, t_2\right]$.

There may be many trust relationships between the subject-subset U_i and the contexts of object V as shown in figure 48. Each edge between subjects-subset U_i and the context V_m may have a different trust metric at the time interval $\left[t_1, t_2\right]$ because the number of subject's rated the each context varies. In this case, trust metrics can be shown by different colors. Now, our trust graph is a colored graph where the trust metric is color. A colored subject-to-object graph is shown in figure 3.9.

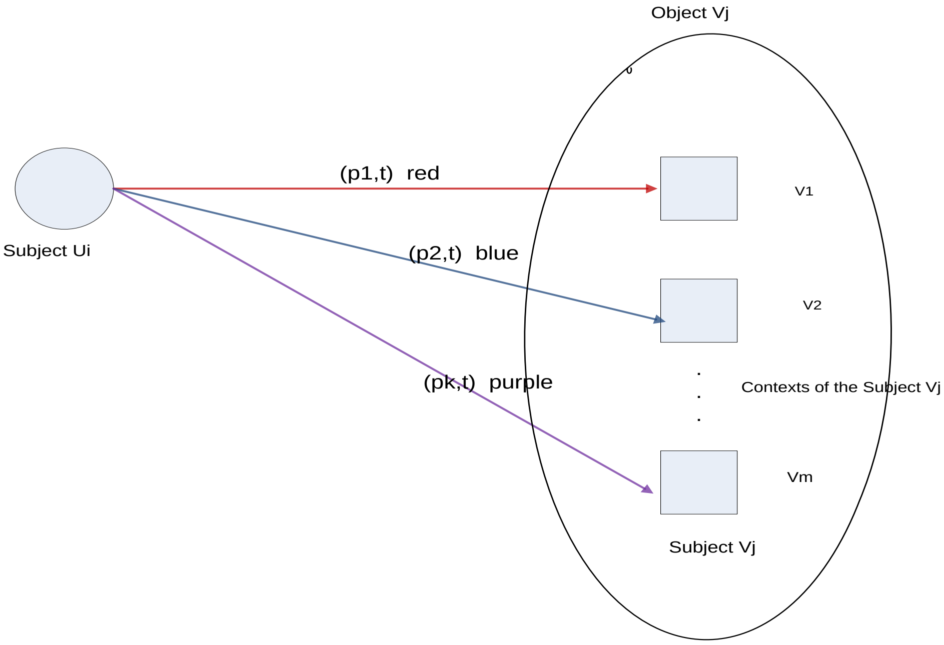


Figure 3.9 A Subject-to-object graph where trust metric is edge color

3.9 Generation of a Colored-Graph Based Trust Model From Real Data

Our model uses the raw-input data for the objects in sub-clusters. We assume that data are created by subjects by using web pages of the system.

Format of the raw input data is as following:

Table 3.1 Raw input-data example

Subject ID	Subject Type	Object ID	Assesment Values												Time Stamp	
			V_1	I	V_2	I	V_3	I	V_4	I	…	…	V_m	I	Day	Time
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm. year	hh.mm:ss
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm. year	hh.mm:ss
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm. year	hh.mm:ss
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm. year	hh.mm:ss
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm. year	hh.mm:ss
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm. year	hh.mm:ss
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm. year	hh.mm:ss
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm. year	hh.mm:ss
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm. year	hh.mm:ss
xxxxxxx	U_i	Name	xxx	xx	xxx	xx	xxx	xx	xxx	xx	…	…	xxx	xx	dd.mm. year	hh.mm:ss
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

Subject Id: Subject Id is a seven-digit number given by the rating system for the raters. Each rater gives his/her personal information and enrolls the system. Each time he/she logs on the system this Id is used.

Subject Type: Each rater defines its rater type before rating. Because rater type is important for the rating assesment. Each rater must select one of the n rater types: U_1, U_2, U_3, …, U_n.

Object Id: Each object recorded in the assesment system has an Id. Our system uses the name of the object as the *object Id. Object Id is given as the 25 alpha numerical characters.*

Assesment Value: Assesment grades are given for the contexts of the object. Each context $V_1, V_2, V_3, ..., V_m$ is rated in the range of integers $[1, k]$. '1' is the lowest grade and *'k'* is the highest grade. For the simplicity maximum grade *k* is defined as 100 in our model.

Importance Value: Each assesment grade is given together by an importance value I. This value represents the importance of the feature for the rater. Importance value is given in range of integers $[1, l]$. 1 is the lowest value and *l* is the highest. If someone does not give an importance value for the context V_m feature it is assumed as *l*. For simplicity maximum *l* value is selected as 10 in our model.

Time Stamp: Time-stamp is the time of the assesment is completed. Time-stamp is given in the <day, time> format. Day is given as <*dd.mm.year*> format. Time is given in the <*hh:mm:ss*> format.

3.9.1 Processing Raw Input Data

By processing raw-input data , "processed-raw input data " is obtained. Importance Values are used to process the Raw-Input Data as follows:

- If I=*l* rating given by the asseser does not change.
- If I$\neq l$ rating given by the asseser is multipilied by (1- 0.2/*l*)

In our model, since the lowest value for *l* is defined as 1, importance value can not be smaller than 0.80.

Processed-raw-input data values are represented as a positive two digit real number by one decimal place like “xx.x”.

Processed-Raw-Input Data is given as follows:

Table 3.2 Processed raw input-data example

Subject ID	Subject Type	Object ID	Assesment Values						Time Stamp	
			V_1	V_2	V_3	V_4	…	V_m	Day	Time
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
xxxxxxx	U_i	Name	xx.x	xx.x	xx.x	xx.x	…	xx.x	dd.mm.year	hh.mm:ss
…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…

By using the processed-raw-input values a*rithmetic-mean values* are calculated of each subject-subset U_n for each context of V_m the *object* V .

Rows of the matrix represent:

$a_{11} - to - a_{1m}$: Subject-subset U_1

$a_{21} - to - a_{2m}$: Subject-subset U_2

$a_{31} - to - a_{3m}$: Subject-subset U_3

…

$a_{n1} - to - a_{nm}$: Subject-subset U_n

Columns of the matrix represent:

$a_{11} - to - a_{n1}$: Subject-subset U_1 's grade for the context V_1.

$a_{12} - to - a_{n2}$: Subject-subset U_2 's grade for the context V_2.

$a_{13} - to - a_{n3}$: Subject-subset U_3's grade for the context V_3.

…

$a_{1m} - to - a_{nm}$: Subject-subset U_n 's grade for the context V_m .

Assesment matrix A_{nxm} for the *object V* is shown in figure 3.10.

$$A_{object.id[t_1,t_2]} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & \dots & a_{1m} \\ a_{21} & a_{22} & a_{23} & \dots & \dots & a_{2m} \\ a_{31} & a_{32} & a_{33} & \dots & \dots & a_{3m} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & a_{n3} & \dots & \dots & a_{nm} \end{pmatrix}$$

Figure 3.10 Assesment matrix example

Definition 3.7 (Elements of Assesment Matrix): Each element a_{ij} of the assesment matrix $A_{object.id[t_1,t_2]}$ represents the rating of subject group U_i for the context V_j. This value is equal to the *"arithmetic mean of the processed-raw input data"* of the subject group U_i for the context V_j in the time interval $[t_1, t_2]$.

For example, rating of subject group U_1 for the context V_3 is a_{13}. This value is equal to the *"arithmetic mean of the processed-raw input data"* of the subject group U_1 for the context V_3 in the time interval $\left[t_1, t_2\right]$.

3.9.2 Adding Weights to Assesment Matrix

In section 3.8 we have made *Assumption 3.1* about trust metric: Trust metric for the context V_m at the time interval $\left[t_1, t_2\right]$ increases as the number of subjects n which rates for this context increases. According to this assumption weights of the ratings given by subject-subsets are not the the same. The number of subjects n rating the context is important for us. The subject-subset U_i which gives the the highest number of assesments is the most valuable. We count from the processed-raw input data and find the number of subjects for each subject-subset:

- n_1 : Total number of raters of the subject-subset U_1.
- n_2 : Total number of raters of the subject-subset U_2.
- n_3 : Total number of raters of the subject-subset U_3.

....

- n_n : Total number of raters of the subject-subset U_n.

Definition 3.8 (Number of Assesers Vector): Number of Assesers Vector N defines the number of elements of each subject-subset U_i and the total number of elements of the subject set U rated for the context V_m at the time interval $\left[t_1, t_2\right]$.

$N_{object.id\left[t_1, t_2\right]} = \left[n_1, n_2, n_3, ..., n_n, m, \left[t_1, t_2\right]\right]$, where $\left[t_1, t_2\right]$ is the time interval in which calculations are made.

$n_1, n_2, n_3, ..., n_n$ are the number of elements of the subject-subsets $U_1, U_2, U_3, ..., U_n$ respectively.

m is the total elements of the subjects set U .

$$m = \sum_{i=1}^{n} n_i$$

Subject-subset with the highest number of assesers will take the highest weight value and the subject-subset with the least number of assesers will take the least weight value.

Definition 3.9 (Assesment-Weight Coefficient): Assesment–weight coefficient determines the weight of assesment of an subject-subset U_i for the context V_j at the time interval $[t_1, t_2]$ is defined as follows:

wU_{ij} *= Assesment-weight coefficient= number of elements* n_i *of the subject- subset* U_i *rated for the context* V_m */*total number elements *m* of the subjects set U rated for the context V_j .

$$wU_1 + wU_2 + wU_3 + ... + wU_n = 1$$

Assesment-weight coeffecients for each subject-subset for the context V_m can be calculated as follows:

wU_{1m} = $0 < n_1 / m < 1$

wU_{2m} = $0 < n_2 / m < 1$

wU_{3m} = $0 < n_3 / m < 1$

…

wU_{nm} = $0 < n_n/m < 1$

Assesment-weight coeffecient is computed in our model as a real number with four decimal places like 0.xxxx .

Definition 3.10 (Assesment-Weight Vector for the Context V_j): Assesment–weight vector determines the weights of assesment of all subject-subsets U_n for the context V_m at the time interval $[t_1, t_2]$ is defined as follows:

$$AWV_{m[t_1,t_2]} = [wU_{1m}, wU_{2m}, wU_{3m}, ..., wU_{nm}]$$

Each context has a different assesment-weight vector.

Trust relationship graph between the sets U and V for the context V_1 by considering the weights shown in figure 3.11. This graph is a *weighted* or a *colored graph* since the weights of the edges are added.

Weights of the edges are shown by colors in the graph.

Rank of weights shown by colors from highest to lowest is shown as follows:

- 1. Color-1 (highest)
- 2. Color-2
- 3. Color-3

…

- n. Color-n (lowest)

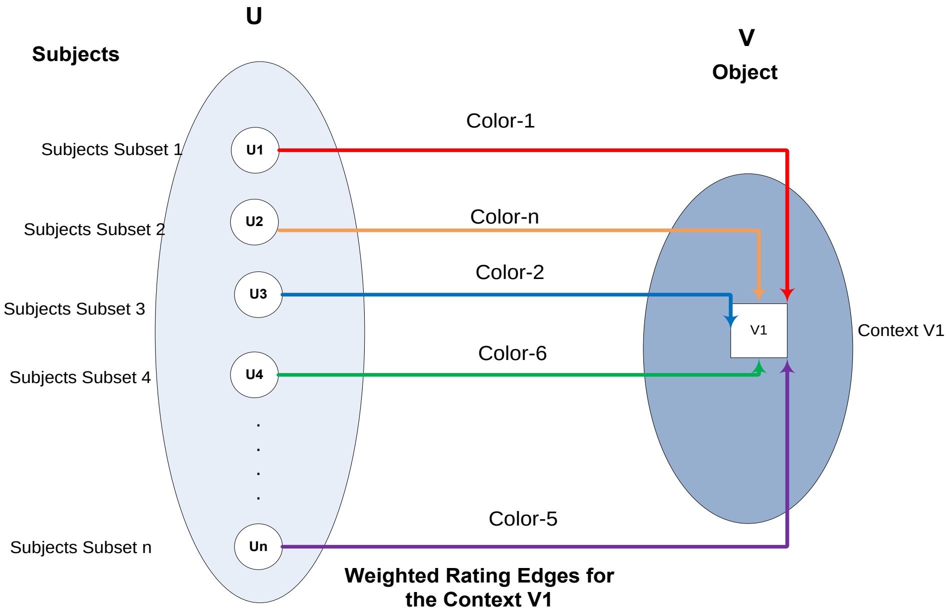


Figure 3.11 Colored trust-graph for the context V_1

3.9.3 Calculation the Popularity Metric for the Contexts

Definition 3.11 (Popularity Metric for the Context V_j): Popularity metric for each context at the time interval $[t_1, t_2]$ is defined as follows:

$$a_{vj} = \left(a_{1j}.wU_{1j} + a_{2j}.wU_{2j} + a_{3j}.wU_{3j} + ... + a_{nj}.wU_{nj}\right)$$

Popularity metric is used to determine the rank of the contexts for each subject group U_i.

In this step, we have an assesment matrix obtained from processed-raw input data

$$A_{object.id[t_1,t_2]} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & \dots & a_{1m} \\ a_{21} & a_{22} & a_{23} & \dots & \dots & a_{2m} \\ a_{31} & a_{32} & a_{33} & \dots & \dots & a_{3m} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & a_{n3} & \dots & \dots & a_{nm} \end{pmatrix}$$

and an assesment-weight coefficient vector for each context.

$$AWV_{m[t_1,t_2]} = [wU_{1m}, wU_{2m}, wU_{3m}, ..., wU_{nm}]$$

Popularity value for the *context* V_1 can be calculated as follows:

$$a_{v1} = (a_{11}.wU_{11} + a_{21}.wU_{21} + a_{31}.wU_{31} + ... + a_{n1}.wU_{n1})$$

Popularity value for the *context* V_2 can be calculated as follows:

$$a_{v2} = (a_{12}.wU_{12} + a_{22}.wU_{22} + a_{32}.wU_{32} + ... + a_{n2}.wU_{n2})$$

Popularity value for the *context* V_3 can be calculated as follows:

$$a_{v3} = (a_{13}.wU_{13} + a_{23}.wU_{23} + a_{33}.wU_{33} + ... + a_{n3}.wU_{n3})$$

…

Popularity value for the *context* V_m : can be calculated as follows:

$$a_{vm} = (a_{1m}.wU_{1m} + a_{2m}.wU_{2m} + a_{3m}.wU_{3m} + ... + a_{nm}.wU_{nm})$$

Now,we have obtained an Assesment-Context Vector ACV for the object V in time-interval $\left[t_1, t_2\right]$. Elements of the ACV are the popularity values calculated for each context in this time interval.

$$ACV_{object.id\left[t_1,t_2\right]} = \left[a_{v1}, a_{v2}, a_{v3}, ..., a_{v_m}, \left[t_1, t_2\right]\right]$$

Time-interval shows the time-gap in which our calculations are made. This can be adjusted by selecting the *initial* t_1 and *final* t_2 values of time. Because our database collects and registers the rater's assesments with a time stamp in the <*dd.mm.year, hh:mm:ss* > format.

3.9.4 Computation of the Popularity and Trust Metrics for the Objects

We have two vectors for the the object V for the same time interval.

1. An assesment-context vector for the object V in time-interval $\left[t_1, t_2\right]$:

   $$ACV_{object.id\left[t_1,t_2\right]} = \left[a_{v1}, a_{v2}, a_{v3}, ..., a_{v_m}, \left[t_1, t_2\right]\right]$$

2. Number of assesers vector N for the object X in time-interval $\left[t_1, t_2\right]$:

   $$N_{object.id\left[t_1,t_2\right]} = \left[n_1, n_2, n_3, ..., n_n, m, \left[t_1, t_2\right]\right]$$

In the same sub-cluster of objects there many other objects.

Objects in the same sub-cluster are shown by V_i. There are a finite number of objects in the same sub-cluster. So, $1 \le i \le j$. Minimum number of objects can be 1 and maximum number of objects can be j. In the same sub-cluster, in the same time interval $\left[t_1, t_2\right]$ there are j assesment-context vectors and j number of assesers vectors.

For the i th object these two vectors can be shown as follows:

$$ACV_{object.id_i[t_1,t_2]} = \left[a_{iv_1}, a_{iv_2}, a_{iv_3}, ..., a_{iv_m}, \left[t_1, t_2\right]\right]$$

$$N_{object.id_i[t_1,t_2]} = \left[n_{i_1}, n_{i_2}, n_{i_3}, ..., n_{i_n}, m_{i_n}, \left[t_1, t_2\right]\right]$$

To find the Popularity-Index pop_i of object in sub-cluster for one of the subject-groups U_i in time interval $\left[t_1, t_2\right]$ following calculation is applied.

For example for the subject-group U_1 popularity of the i th object:

$$pop_i = n_{i_1} / \sum_{i=1}^{j} n_{i_1} \left[a_{iv_1} + a_{iv_2} + a_{iv_3} + ... + a_{iv_m}\right]$$

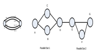
For the sub-cluster for the subject-group U_1, j popularity values will be found. These are: pop_1, pop_2, pop_3, …, pop_j.

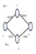
Since we have defined trust value in the real numbers interval $\left[0,1\right]$ a normalization is required. This can be done as follows:

pop_i value can be maximum *m. (max-value of*) a_{iv_m}. For example if a scale of *1 to 5* is selected maximum pop_j value can be *mx5.*

$trust_i$ is defined as:

$trust_i$ = pop_i/ *m. (max-value of*) a_{iv_m}

For the sub-cluster for the object U_1 j value of trust metric will be found. These are:  , $trust_2$, $trust_3$, …, $trust_j$.
By ordering the trust values from the largest numerical value to the smallest numerical value the most trusted object for the subject-group U_1 is found.

For each the subject-group  to U_n same calculations are repeated.

To find the *Overall-Popularity Index* $opop_i$ of an object in its sub-cluster for all of the subject-groups included U_n $1 \le i \le n$ in time interval $[t_1, t_2]$ following calculation is applied.

$opop_i$ of the object i in time interval $[t_1, t_2]$:

$$opop_i = m_i / \sum_{i=1}^{j} m_{i_n} \; . \; \left[a_{iv_1} + a_{iv_2} + a_{iv_3} + ... + a_{iv_m} \right]$$

For all subject-groups included, j overall-popularity values will be found for the objects in the sub-cluster. These are: $opop_1$, $opop_2$, $opop_3$, …, $opop_j$.
Since we have defined trust value in the real numbers interval $[0,1]$ a normalization is required. This can be done as follows:

$opop_i$ value can be maximum *m. (max-value of)* a_{iv_m}. For example if a scale of *1 to 5* is selected maximum pop_j value can be *mx5*.

Overall-Trust $otrust_i$ for the object i in time interval $[t_1, t_2]$:

$otrust_i =$ $opop_i$ */ m. (max-value of)* a_{iv_m}

For all subject-groups included, j overall-trust values will be found for the objects in the subcluster. These are: $otrust_1$, $otrust_2$, $otrust_3$, …, $otrust_j$.

By ordering the overall-trust values from the largest numerical value to the smallest numerical value the most trusted object for the all subject-groups included is found.

*Assumption 3.3:*Based on central limit theorem (Neumann. 2000), if number of assesers for each subject-subset U_{ij} for the context V_j $n \geq 30$ in time interval $\left[t_1, t_2\right]$ assesers grades are assumed normally distributed.

Definition 3.12 (Confidence probability α of the $otrust_i$): Confidence probablity α of the $otrust_i$ at the time interval $\left[t_1, t_2\right]$ is defined as follows:

$$P\left(\mu - z_\alpha \frac{\sigma}{\sqrt{n}} \leq otrust_i \leq \mu + z_\alpha \frac{\sigma}{\sqrt{n}}\right) = \alpha$$

where:

- α (Alpha): Confidence probability
- μ (Arithmetic mean): Arithmetic mean of grades of assesers of the subject-subset U_{ij} for the context V_j $n \geq 30$ in time interval $\left[t_1, t_2\right]$ (grades can be weighted).
- σ (Sigma): Standart deviation of assesers grades.
- n: number of assesers for each subject-subset U_{ij} for the context V_j), n can not be smaller than 30 according to assumption -3.
- z_α: z_α value can be found from Table-3 according to the chosen α value.
- *otrust*: Overall trust value.

Definition 3.13 (Confidence interval of the $otrust_i$ with α probability): Confidence interval of the $otrust_i$ with α probability at the time interval $\left[t_1, t_2\right]$ is defined as follows:

$$\left(\mu - z_\alpha \frac{\sigma}{\sqrt{n}}, \mu + z_\alpha \frac{\alpha}{\sqrt{n}}\right) \text{ where } \mu = otrust_i .$$

Definition 3.14 (Arithmetic Mean of Weighted Data): If each member of a set is multiplied by a constant c, then the mean μ will be c times of its value before the constant was multiplied.

Definition 3.15 (Variance of Weighted Data): If each member of a set is multiplied by a constant c, then the standard deviation σ will be |c| times of its value before the constant was multiplied.

Our contributions are importance value and calculation of total trust as real number intervals in the range of $\left[0,1\right]$ by using confidence probability. Our model is so flexible and can be applied to any kind of survey easily. Results can be used in the comparison of the performance of the organization with itself or its competitors. Flexibility of our model is shown in the chapter 5.

CHAPTER FOUR
TRUST PROPOGATION MODELING

4.1 Motivation

How much should you trust the friend of a friend? This question is the basis of the trust transitivity problem. It is clear that trust is transitive to some extent. Many people use their friend's opinions about others to some degree if they have no a direct trust relationship with them. But, everyone does not use the same rules to assess his/her friend's opinions into our their assessments. Whatever the transitivity rules a person uses, the concept of trust that people actually use, allows others to use their friend's opinions. Trust values obtained from different paths may be different because people may have different opinions about the same friend. Personal trust is relative, and depends on personal perspective.

A good outcome for one person could be a bad outcome for another. Trust might not be equal in both parallel paths.The problem is how to calculate the total propogated trust value of the parallel-serial chain. Selection of the method depends on the trust policy we use (Orgun and et al., 2006).

The chapter also contains a novel algorithm for calculation of confidence propogation which does not exist in similar reseaches. Trust propogation models that we select and propose are given in the following subsections.

4.2 Serial and Parallel Chains for Trust Propogation

A serial trust chain can be explained by a simple example. If person A trusts person B who trusts person C, then A trusts C. This assumes that B tells A he/she trusts C. This is called recommendation. In real life trust is not always transitive. For example, person A trusts B as a good teacher, and B trusts C as an experienced doctor, does not imply that A trusts C. However, under certain contexts (Josang et al, 2005), trust can propogate and a serial trust chain can be used to derive trust propogation.

Let us assume that A needs an experienced doctor and asks B for him/her advice. B is trusted by A to know about an experienced doctor. B in turn trusts C to as an experienced doctor and tells him/her honest opinion about C to A. This situation is illustrated in figure 4.1, where indexes indicate the order in which the trust relationships and recommendations are formed. The opinion of A about C as an experienced doctor is propogated trust. The context of the trust is to be an experienced doctor.

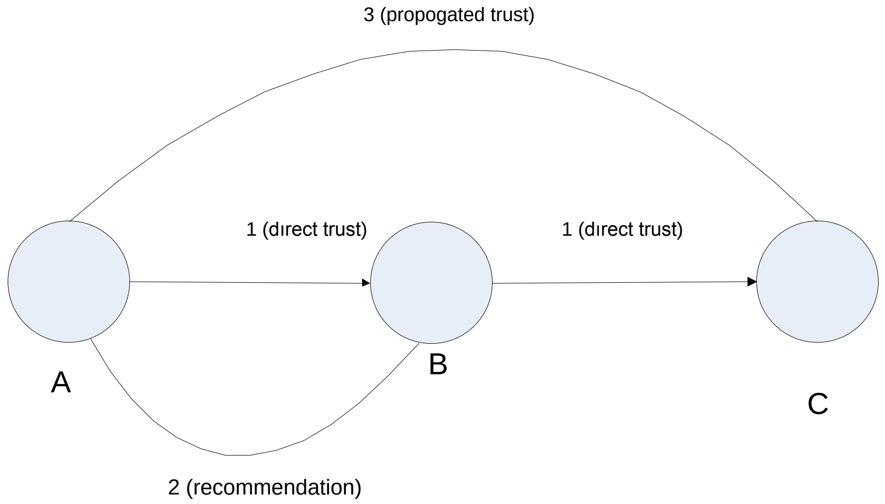


Figure 4.1 Serial trust chain

To be beter informed when making decisions, we try to collect recommendations from severel sources in order. This can be modelled as paralel trust combination. It is illustrated in figure 4.2 where again the indexes indicate the order in which the trust relationships and recommendations are formed.

Let us assume again that A needs an experienced doctor and he/she asks B to recommend an experienced doctor. B recommends his/her good friend D. A would like to get a second opinion, so he/she asks C whether he/she heard about D. C also knows and trusts D. If both B and C recommend D as a good doctor, A’s trust in D will be stronger than if she had only asked B. Parallel combination of positive trust has the effect of strenghtening the propogated trust (Josang et al, 2003). If A receives conflicting recommended trust, e.g. trust and distrust at the same time, A needs some method for combining these conflicting recommendations in order to derive his/her trust in D.

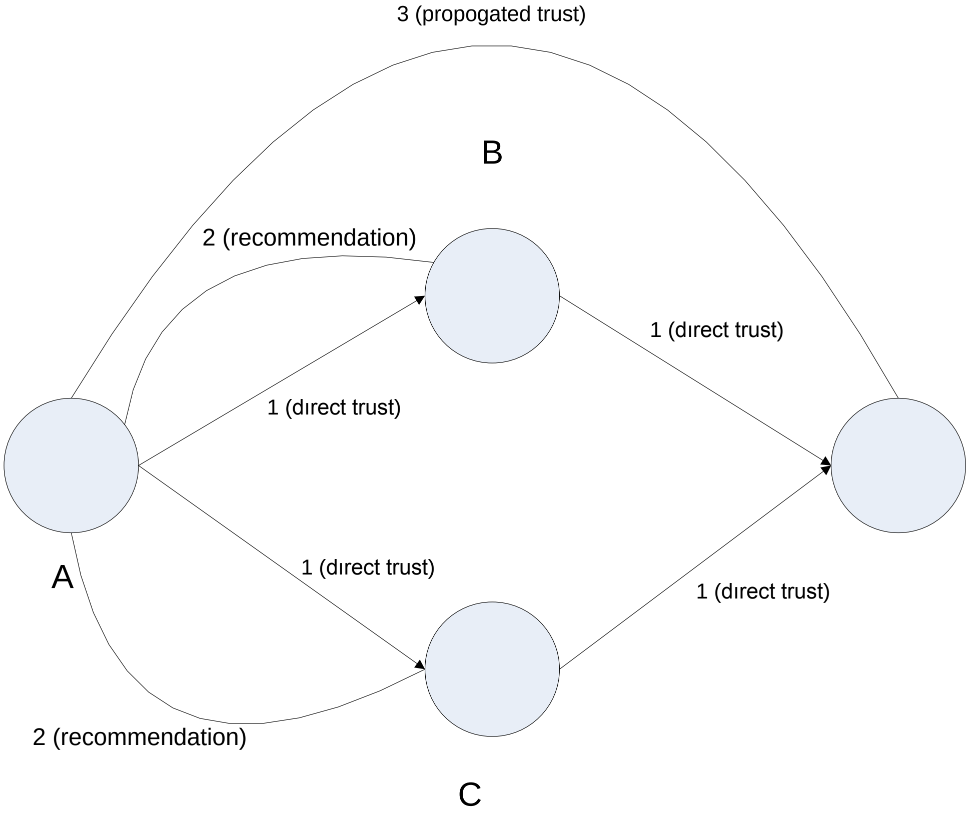


Figure 4.3 Parallel trust chain

4.3 Trust Propogation in Serial Trust Chains

Definition 4.1 (Propogated Trust Value for a Serial Trust Chain): Propogated trust value for the n vertex serial trust chain is equal to the multiplication of assigned trust values of the edges.

$i_T = \prod_{i=1}^{n-1} p_i$, where p_i is the assigned trust value of the i th edge.

Definition 4.2 (Propogated Confidence Value for a Serial Trust Chain): Propogated confidence value for the n vertex serial trust chain is equal to the multiplication of assigned confidence values of the edges. Confdence value θ_i is a real number in the interval $\left[0,1\right]$. In our work confidence values below 0.8 are not considered.

θ_T = $\prod_{i=1}^{n-1} \theta_i$, where p_i is the assigned confidence value of the i th edge.

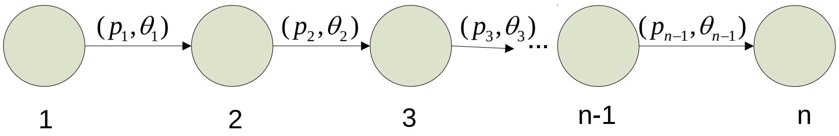


Figure 4.3. A *n* vertex serial trust chain

As can be seen from the definition, as long as the chain propogated trust and confidence values becomes smaller.

4.4 Trust Propogation in Parallel Trust Chains

Definition 4.3 (Propogated Trust Value for a Parallel Trust Chain): Propogated trust value for the k transitive path parallel trust chain is equal to the mean value of assigned trust values of the transitive paths.

i_T = $\frac{1}{k} \sum_{k=1}^{m} p_i$, where k is the total number of parallel transitive paths and p_i is the assigned trust value of each transitive path.

Definition 4.4 (Propogated Confidence Value for a Parallel Trust Chain): Propogated confidence value for the k transitive path parallel trust chain is equal to the mean value of assigned confidence values of the transitive paths.

$\theta_T = \frac{1}{k}\sum_{k=1}^{m}\theta_i$, where k is the total number of parallel transitive paths and θ_i is the assigned confidence value of each transitive path. Confdence value θ_i is a real number in the interval $[0,1]$. In our work confidence values below 0.8 are not considered.

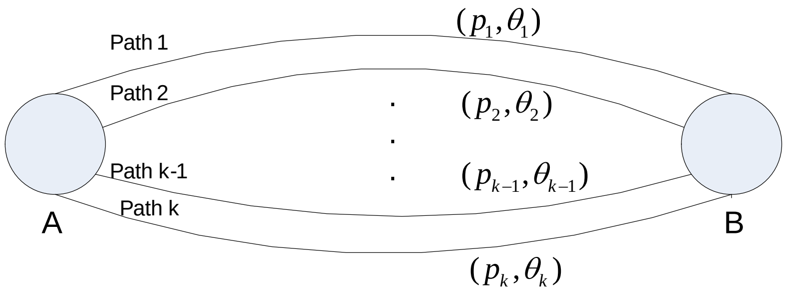


Figure 4.4 A two vertex *k* path parallel trust chain

There are methods for selecting the maximum or minumum values on paralel paths depending on the optimistic or pessimistic approach respectively (Chen and et al., 2009). This approach only considers trust values. Confidence values are not involved.

We calculate the mean value of the trust and confidence values of transitive paralel chains to assess the propogated trust between two vertice. We propose that this method is more fair compared to optimistic and pessimistic approaches.We also calculate the propogation of confidence values of trust chains which is neglected in similar researches.

4.5 Trust Propogation in Combined Serial-Parallel Chains

Long trust chains may be composed of serial and paralel paths together. The method to calculate the propogated trust as follows:

- Step1: Reduce each paralel path to a single path by using the definitions 6 and 7. A serial equivalent chain is obtained.
- Step 2. Calculate the propagated trust over the serial chain by using the definitions 4 and 5.

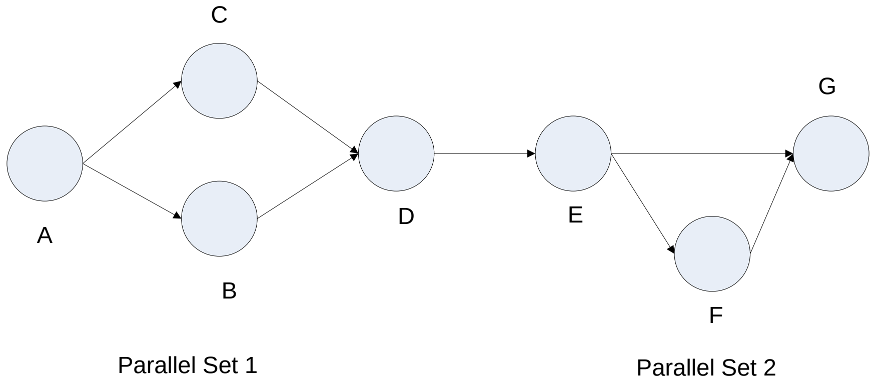


Figure 4.5 A combined serial parallel trust chain before reduction

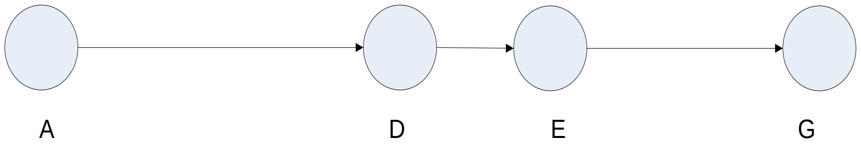


Figure 4.6 Reduced equivalent trust chain of figure 4.5

As we highlighted before long trust chains may produce small trust and confidence values which could be meaningless. To prevent situations alike some additional methods are proposed in the literature and summarized in the following subsections.

4.5.1 The Least Strongest Link of the Trust Chain

It is known in the real life that a chain can not be stronger than its weakest link. Similarly, a long trust chain can propogate the trust equal to the smallest trust value of the link (Theodorakopoulos and et al., 2006). This method considers only trust values. That means: only trust values of the links are compared, confidence values are neglected.

Steps to calculate the propogated trust by this method is as follows:

- Step1: Reduce each paralel path to a single path by using the definitions 6 and 7. A serial equivalent chain is obtained.
- Step 2. Calculate the propagated trust over the serial chain by using the definitions 4 and 5.
- Step 3. Propogated trust is equal to the smallest trust value of the links. Confidence values are neglected.

This method is useful for long chains since very small propogated trust values are not taken into account.

4.5.2 Confinment of the Number of Vertice for Trust Propogation

Effect of vertice for trust propogation decreases as number of vertice increases.Far vertice has very small effect and may be totally neglected (Theodorakopoulos et al, 2006). Recommendations of near vertices are more valuable for us. For example, we can confine n=4 for the serial chain in figure 4.7. Calculated propagated trust will have a much higher value than calculated for n vertice and will be more meaningful for us.

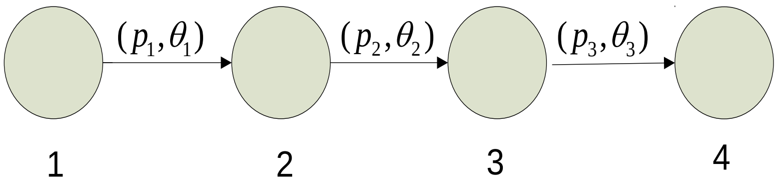


Figure 4.7 Confined serial trust chain for *n*=4

4.6 Numerical Trust Propogation Examples

Trust is transitive to some extent and many people use their friend's opinions about others to some degree. But, everyone does not use the same rules to incorporate our friend's opinions into our own judgments. Whatever the transitivity rules a person uses, the concept of trust that people actually use, allows others to use their friend's opinions.

As an example let's consider the following simple serial trust chain(subject to subject graph). Person C asks her friend B's recommendations about the Hotel Basmane. Person B has no direct experience with Hotel Basmane and he read an article the Hotel Basmane on a serious magazine A on the internet. B evaluates his opinion about the Hotel Basmane according to the article he read. The trust value of the magazine about the Hotel is 0.9 with a confidence 0.9. B tells to C his recommendation about the hotel as 0.8 with a confidence value 0.9. What is total transitive trust value about Hotel Basmane at time t for the given trust values?

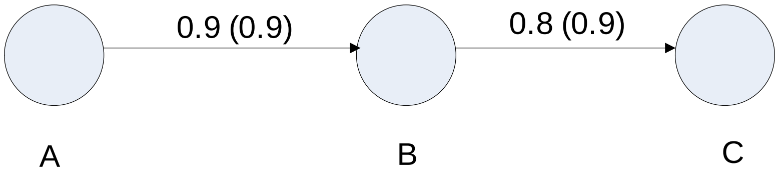


Figure 4.8 A Simple transitive trust chain

Let us define the total transitive trust value tranferred to the person C as the multiplication of direct trust values between A to B and B to C. Let us assume trust values between 0 and 1. Then the value of the final recommendation is 0.9 x 0.8= 0.72 with a confidence value 0.9 x 0.9 =0.81.

Similarly for a *longer serial trust chain* the value of the final recommendation for the person E about the Hotel Basmane at time t can be computed as follows:
0.9 x 0.6 x 0.6 x 0.5 = 0.162 with a confidence value 0.95x0.9x0.85x0.8=0.58

4.6.1 Trust Propogation in Serial and Parallel Trust Chains

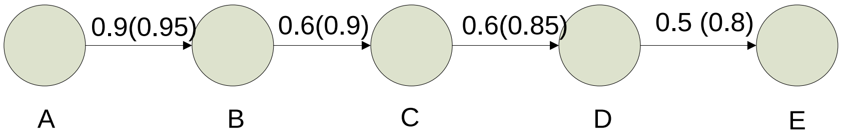


Figure 4.9 A longer transitive trust chain

In the case of a parallel trust chain the value of the final recommendation can be computed as follows:

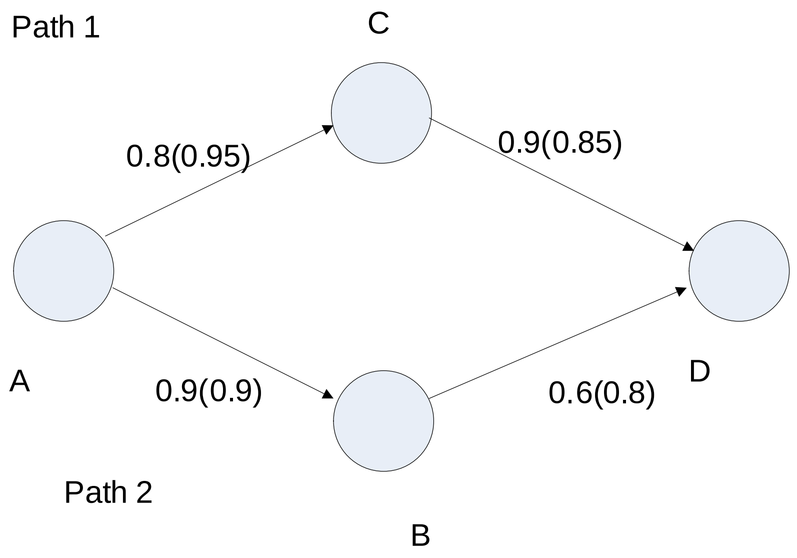


Figure 4.10 A sample transitive trust chain

It can be seen from the figure 4.10, graph is composed of two transitive serial paths. Trust values of transitive serial paths can be computed seperately.

Path 1: 0.8 x 0.9 = 0.72 with confidence value 0.95x0.85=0.81

Path 2: 0.9 x 0.6 = 0.54 with confidence value 0.9x0.8=0.72

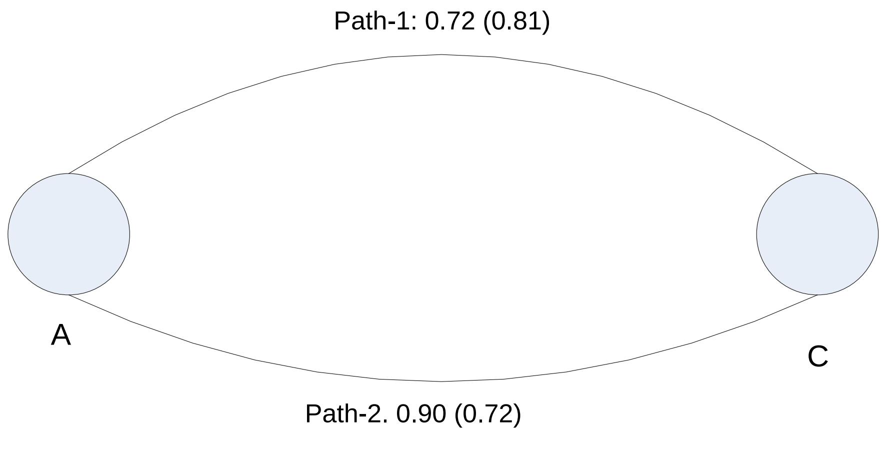


Figure 4.11 Two parallel paths example

Trust values obtained from different paths may be different because people may have different opinions about the same hotel. Personal trust is relative, and depends on personal perspective. A good opinion for one person could be a bad opinion for another. Trust might not be equal in both parallel paths.The problem is how to calculate the total recommendation value of the parallel-serial chain. Selection of the method depends on the trust policy we use. We choose the mean value of the transitive serial chains. Then the recommendation value of the parallel-serial chain can be computed as follows:

i_T = ½ x (0.72+0.54) = 0.63 with confidence value ½ x (0.81+0.72) = 0.77

A more complicated example for a parallel-serial trust chain can be given as follows:

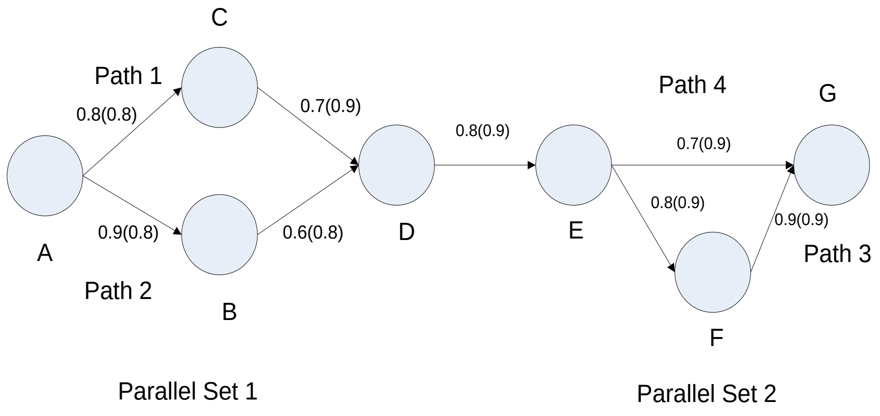


Figure 4.12 A more complicated transitive trust chain

It is seen on figure 4.12, there are two parallel transitive sets.

Parallel Set 1: path 1 and path 2

Parallel Set 2: path 3 and path 4

Propogated trust value for parallel set 1: $\frac{1}{2}x\left[\left(0.8x0.7\right)+\left(0.9x0.6\right)\right]$= 0.55

Propogated confidence value for parallel set 1: $\frac{1}{2}x\left[\left(0.8x0.9\right)+\left(0.8x0.8\right)\right]= 0.68$

Propogated trust value for parallel set 2: $\frac{1}{2}x\left[0.7+\left(0.8x0.9\right)\right] = 0.71$

Propogated confidence value for parallel set 2: $\frac{1}{2}x\left[0.9+\left(0.9x0.9\right)\right] = 0.86$

Our graph is now converted to a serial trust chain shown as the following:

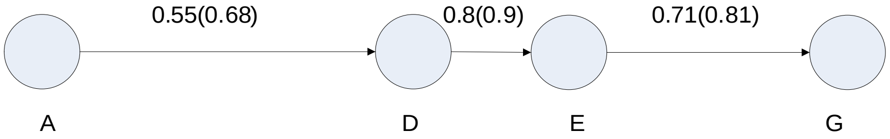


Figure 4.13 Converted transitive trust chain.

Total propogated trust value can be calculated : $\left(0.55x0.8x0.71\right)=0.31$

Total propogated confidence value can be calculated : $\left(0.68x0.9x0.81\right)=0.496$

4.6.2 The Least Strongest Link of The Chain

It is known in the real life that a chain can not be stronger than its weakest link. Similarly, a long trust chain can propogate the trust equal to the smallest trust value of the link (Theodorakopoulos et al, 2006). This method considers only trust values. That means: only trust values of the links are compared, confidence values are neglected.

For the example given in figure 4.13 weakest link of the chain has the trust value of 0.55. Then the propogated trust value over this chain is only 0.55.

4.6.3 Confinment of the Number of Vertice for Trust Propogation

Effect of vertice for trust propogation decreases as number of vertice increases.Far vertice has very small effect and may be totally neglected (Theodorakopoulos et al, 2006). Recommendations of near vertices are more valuable for us. For example, we can confine n=4 for the serial chain in figure 4.14. Calculated propagated trust will have a much higher value than calculated for n vertice and will be more meaningful for us.

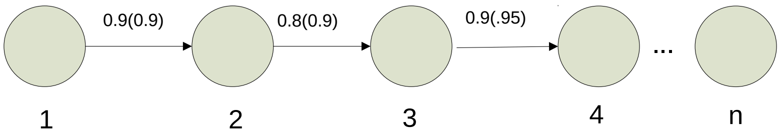


Figure 4.14 Confined serial trust chain for *n*=4

Total propogated trust value can be calculated for figure 4.14:

$(0.9x0.8x0.9) = 0.648$

Total propogated confidence value can be calculated for figure 4.14:

$(0.9x0.9x0.95) = 0.780$

In the proposed trust propogation model in serial-parallel trust chains we have two main contributions. In paralel chains we calculate the arithmetic mean value of trust values for paralel paths. This propery does not exist in previously proposed min-max algorithms. In serial and paralel chains we calculate confidence value propogation. This feature does not exist in any of the similar reseaches.

4.7 Service Oriented Trust Propogation

So far we have investigated trust propogation for human to human interactions. Web-based environments typically span interactions between humans and software services. There are many cases where web-based services interact with other web-based services automatically (Şerif et al, 2010). Many users using many web-sites and these web-sites are using automatically other web-sites as shown in figure 4.15.

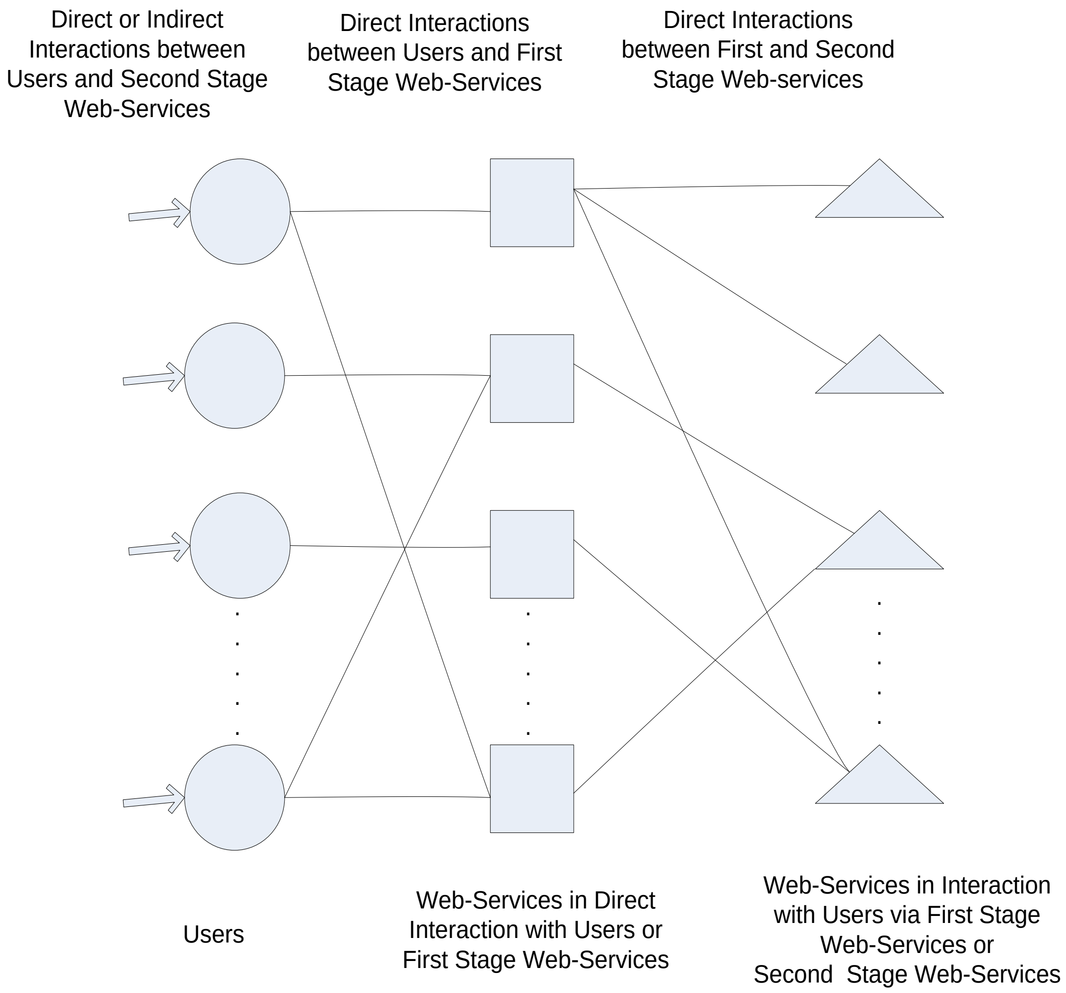

Direct or Indirect Interactions between Users and Second Stage Web-Services
Direct Interactions between Users and First Stage Web-Services
Direct Interactions between First and Second Stage Web-services
Users
Web-Services in Direct Interaction with Users or First Stage Web-Services
Web-Services in Interaction with Users via First Stage Web-Services or Second Stage Web-Services

Figure 4.15 Web-service interactions first &second level

This network is a mesh network. Establishment of a trust relationship between a user and a first stage web-service depends on combination of user’s direct experiences and recommendations between second stage web-services and the direct experiences between the first stage web-service and the second level web-services. For example consider the service e-government web-service in Turkey. This web service is in automatic interaction with many other goverment web-services and the user may have or not direct interactions and/or recommendations from second stage web-services. This case is shown in figure 4.16. There are two cases that should be investigated.

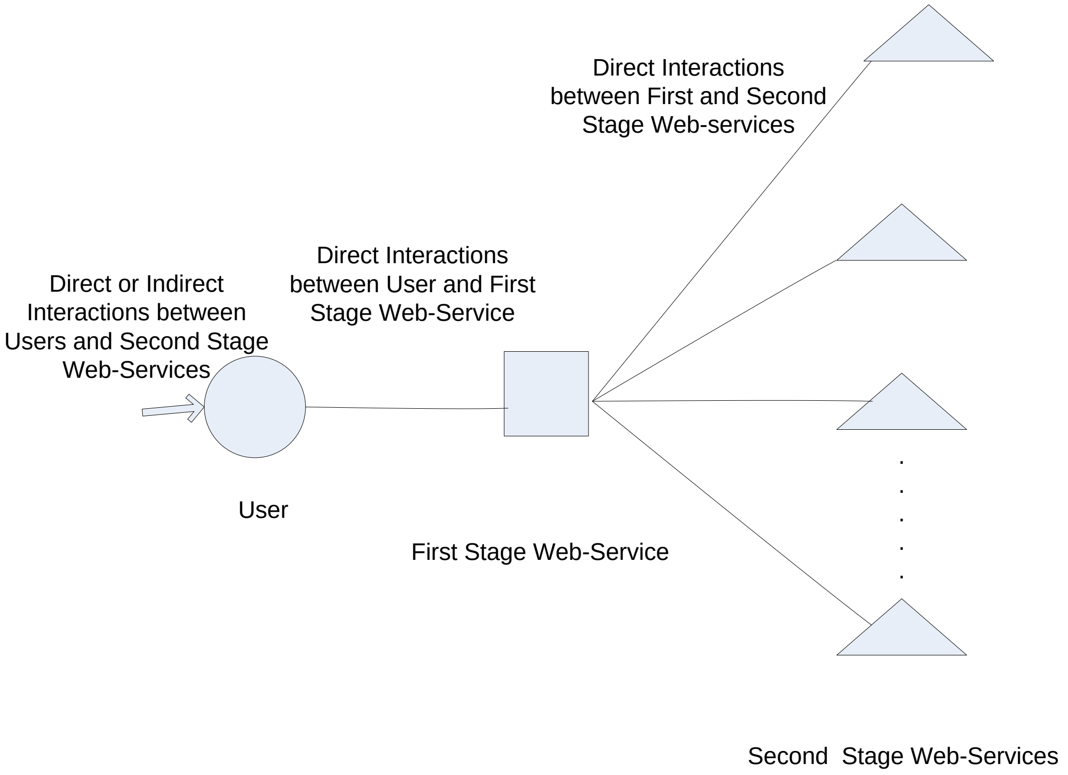

Direct Interactions between First and Second Stage Web-services
Direct Interactions between User and First Stage Web-Service
Direct or Indirect Interactions between Users and Second Stage Web-Services
User
First Stage Web-Service
Second Stage Web-Services

Figure 4.16 User interaction with first and second stage web-services

The graph in figure 4.16 is a object to subject bipartite graph. Trust of the user to the second stage web-services depends on direct interactions or recommendations about them. We do not consider how trust formed between user and second stage web-services in the history.

Assumption 4.1: Trust of the user about each second stage web-service depends on history. How it is formed is not our interest. User's trust value to second stage web-service in time interval $(-\infty, t]$ is shown as T_{UWS2_i} where: $i = 1, 2, ..., n$; n is the number of the second stage web-services in direct interaction with first stage web-services in the history.

Assumption 4.2: Trust of the user about each second stage web-service formed in history has only one of the following two values:

- *Unknown: Unknown* means user has no idea about that second-stage web-service. It is shown as variable *u.*
- *A Real Number:* T_{UWS2_i} has a real number value in the $[0,1]$ interval. This value shows the trust of the user about each second stage web-service formed in the history. *0* means no trust and *1* means full trust.

Trust between first and second stage web-services also depends on direct interactions between them in history. Situation is different than the former case. Former case defines an direct/indirect interaction between a human and a service. At the present case we define a service to service interaction.

Definition 4.5: Trust value formed between stage-1 and stage-2 web-services in time interval $(-\infty,t]$ is the number of succesful interactions/number of total interactions between two web-services.

$T_{WS1-i} = ns_{1-i} / nt_{1-i}$

where $i = 1,2,...,n$; n is the number of the second stage web-services in direct interaction with first stage web-service. *t* shows the present time.

For example, if there are total 558 interactions between stage-1 web service and the stage-2 web service numbered as 1. Number of successful interactions in the history is 402. Trust value is calculated by using definition 4.5.

T_{WS1-1} = 402/558=0.7204

Assumption 4.3: Trust value formed between web-services in time interval $(-\infty,t]$ does not have an unknown value *u*. That means first-stage web service never interacts with an unknown second-stage web-service.

Definition 4.6: First stage service to second stage service trust weight factor in time interval $\left(-\infty, t\right]$ is the total number of transactions between service-1 and service-2/ total number of transactions of service-1 with all second stage services.

$$TW_{S1-i} = nt_{1-i} / \sum_{i=1}^{n} nt_{1-i}$$

where $i = 1, 2, ..., n$

The problem is to calculate the trust value of the user on the first-stage web-service which is in direct interaction between second stage web-services.

Definition 4.7: A user's trust in time interval $\left(-\infty, t\right]$ to web-service stage-1 which uses the web-service stage-2 is:

$$T_{UWS1_i} = [\ T_{UWS2_i} + T_{WS1-i}\ ] / 2$$

where $i = 1, 2, ..., n$

If a user's trust in time interval $\left(-\infty, t\right]$ to web-service stage-2 is unknown then trust of the user to the web-service-1 is:

$$T_{UWS1_i} = T_{WS1-i}$$

where $i = 1, 2, ..., n$

That means unkown value is neglected and only trust value between two web-services considered.

Definition 4.8: A user's total trust in time interval $\left(-\infty, t\right]$ to web-service stage-1 which uses n web-services of stage-2 is:

$$T_{UWS1T} = \sum_{i=1}^{n} [\ T_{UWS1_i} \text{ x } TW_{S1-i}\ ]$$

Assumption 4.4: Confidence value of trust about each second stage web-service depends on history. How it is formed is not our interest. Confidence value of trust to the second stage web-service in time interval $(-\infty, t]$ is shown as θ_{UWS2_i} where: $i = 1, 2, ..., n$; n is the number of the second stage web-services in direct interaction with first stage web-services in the history. θ_{UWS2_i} is a real number in the interval $[0,1]$. In our work confidence values below 0.8 are not considered.

Assumption 4.5: If trust value of the user to the second stage web-service is unkown than confidence value does not exist.

Definition 4.9: Confidence value of trust in time interval $(-\infty, t]$ to web-service stage-1 which uses the web-service stage-2 is:

$$\theta_{UWS1_i} = [\ \theta_{UWS2_i} + \theta_{WS1-i}\ ]/2$$

where $i = 1, 2, ..., n$

If a user's trust in time interval $(-\infty, t]$ to web-service stage-2 is unknown then user's confidence to web-service stage-1 is:

$$\theta_{UWS1_i} = \theta_{WS1-i}$$

where $i = 1, 2, ..., n$

That means unkown value is neglected and only confidence value between two web-services considered.

Definition 4.10: Total confidence in time interval $(-\infty, t]$ to web-service stage-1 which uses *n* web-services of stage-2 is:

$$\theta_{UWS1T} = \sum_{i=1}^{n} [T_{UWS1_i} \text{ x } \theta_{UWS1_i}]$$

This definition uses trust weight vectors of each second-stage web-services for total confidence calculation.

4.7.1 Discussion About Service Oriented Trust Propogation

Service oriented trust propogation is a hot topic in computer science. Christopher J.E. et al (2009), propose a model to ases the trustworthiness of other agents based on automated transactions in commerce. They claim that agents with high measured discount factors often behave in a trustworthy manner. They offer a mathematical model that discount factors is a measure of trustworthiness.

Zia A. T. (2008), proposes a framework which uses the reputation and trust management to detect trust behaviour, on the basis of the responses from other neighbouring nodes. If the number of trust entries concerning a particular node reaches a treshold, that node is declared as *untrusted* node.

Yang Y. et al (2002), defines a method depends on measuring relevance between services to calculate trust value. They compare common relevance attributes with other unused attributes and calculate a trust value.

Our model calculates the trust value between first and second level services on the success of automated transactions. A second level service with a high number of successful transactions is considered more trustworthy. The main contribution of our model is to consider confidence value propogation between services. Confidence value is completely neglected in similar researches.

CHAPTER FIVE

TRUST ASSESMENT CASE STUDIES

5.1 Motivation

What is the aim of using *trust*? We use trust to deal with *risks.* Risks depend on the actions of others. In a perfect world , we do not need to trust anybody. If there are no risks trust is not required (Marsh et al, 2005). If everyone is completely trustworthy, there is no risk associated to the behavior of others.

Estimating trust from direct experience is not straightforward. Because some services does not directly give details of their composition to their consumers. A consumer may interact with a composed service without knowing about the services that underlie it. In such a case, evaluating the trustworthiness of a service is not easy. For example, a consumer books an itinerary from a composed travel agent service, which interacts with other underlying services like flight services, hotel services, and transportation services (Michalakopoulos et al, 2005). Suppose the consumer is not satisfied with the composed service because of its late response time. The model should penalize the composed service, as well as some of the underlying ones.

If the hotel service, for example is reported to be the cause of an unsatisfactory quality value, the model should reflect the changes in the way that consumers or other composed services would become reluctant to interact with it. Also, as the amount of experience of the rater increases, the model should be able to suggest superior compositions. Our trust model we design aims to help the consumers to make the *realistic decisions*.

In the following two case studies we show the usage of our contributions:

- Using importance value to discriminate user class preferences.
- Calculation of trust in selected time intervals.

- Calculation of total trust as real number intervals in the range of $[0,1]$ by using confidence probability.

In the third case study, we show an application of our main contribution of our model: To consider evaluate confidence value propogation between services. Confidence value is completely neglected in similar researches.

5.2 Hotel Trust Assesment System

In this scenario we will try to estimate trust relationships for helping decision of customers. Bipartite graphs will be used for modelling. A bipartite graph is a graph where nodes can be divided into two seperate groups U and V such that no edge connects the vertices in the same group. In our model we have two sets of entities:

U: (Subjects) or Raters who rates the hotel they stayed during their trip. Raters set is composed of five subsets:

- Business Reviewers Subset
- Couples Reviewers Subset
- Family Reviewers Subset
- Friends Reviewers Subset
- Solo Travel Reviewers Subset

V (Objects) or Rated-Entities: Hotel X features rated by the customers.

Elements of the set V are real values representing the rater's grades for the features of the hotel.

- Rater's grade for Value
- Rater's grade for Rooms
- Rater's grade for Location

- Rater’s grade for Cleanliness
- Rater’s grade for Service
- Rater’s grade for Sleep Quality

The relationship between these two groups is reflected by the edges.

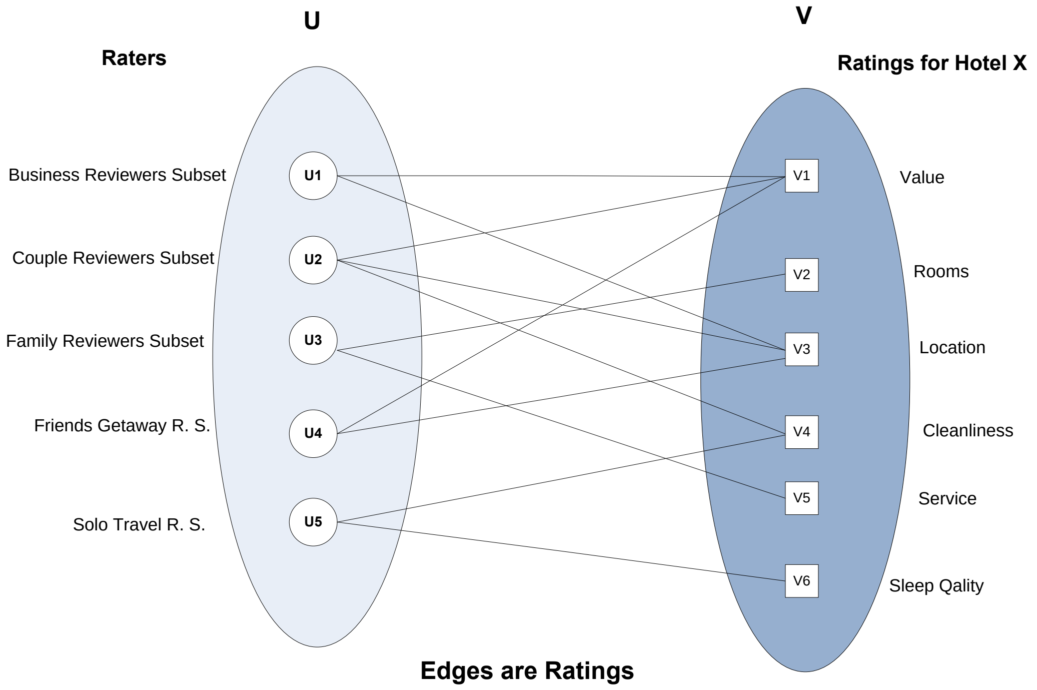


Figure 5.1 Bipartite graph modelling of hotel trust assesment system

5.2.1 Modelling Hierarchy of Clusters

We begin describing the layout of the model as a bipartite graph of objects and users. But the real model is not so small. For example, Hotel Konak is only a single hotel in the city of Izmir. There are many others also in that city. If we want the learn the ranking of raters for Izmir hotels what should we do? Obviously, we need a larger and hierarchical model.

5.2.2 Tree-like Structure of Clusters

In the hierarchical procedures, we construct a hierarchy or tree-like structure to see the relationship among entities. In our example, entities are clusters of hotels. Root is the largest cluster containing all hotels in the world. Sub-clusters are Europe, Asia, America, Africa and Australia hotels. Turkey hotels is a sub-cluster of Europe hotels.

Sub-clusters of Turkey hotels are city hotels like İzmir, İstanbul, etc. Sub-clusters of Izmir hotels are the hotels of the division of the city like Konak, Alsancak, Basmane, etc. Hotel X is the leaf of the sub-cluster Konak Hotels. Tree-like structure of world hotels is shown in figure 5.2.

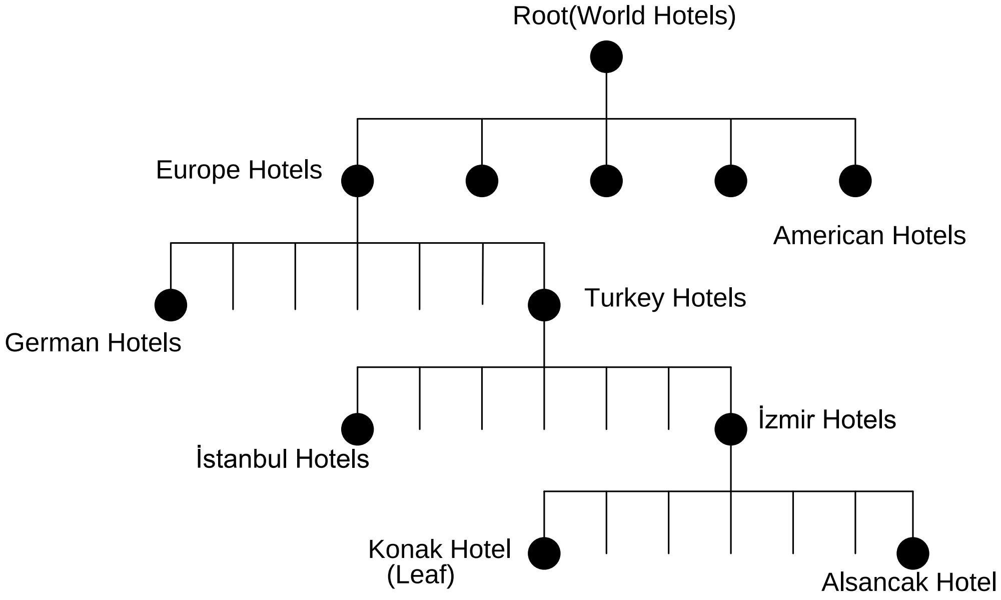


Figure 5.2 Tree-like structure of clusters

Tree-like structure of the word hotels can be also shown as a Wenn-diagram as shown in figure 5.3.

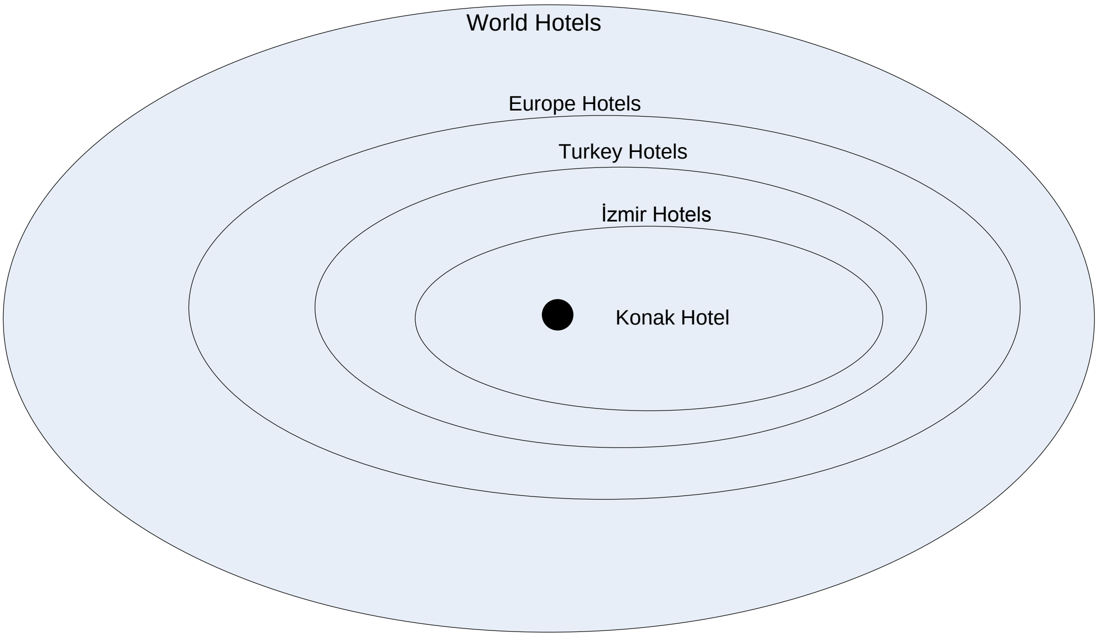


Figure 5.3 Wenn-diagram representation of hierarchy

5.2.3 Raw-Input Data

Hotel trust assesment case study is based on some hypothetical raw-input data for three hotels in Izmir sub-cluster. Raw input-data example is shown in table 5.1.

Raw- input data example contains following data:

Table 5.1 Raw input-data example

User ID	User Type	Hotel ID	Assesment Values												Time Stamp	
			Val	I	Ro	I	Lo	I	Cl	I	Ser	I	Sl	I		
0011183	Business	Konak	5	3	4	2	3	3	4	3	8	3	3	3	12.04.2011	18.00:30
0012185	Business	Alsancak	4	3	3	3	4	2	2	2	2	3	3	1	11.04.2011	15:00:00
0010009	Family	Konak	3	3	4	2	3	3	4	2	4	1	4	3	10.04.2011	16:00:04
0015143	Friends	Konak	3	2	4	1	4	3	4	1	3	2	3	3	09.04.2011	14.00:06
0022145	Solo	Basmane	4	3	4	2	3	3	4	2	4	3	4	3	08.04.2011	18:23:24
0019653	Business	Alsancak	3	1	4	2	4	2	3	1	3	2	3	2	07.04.2011	09.03:05
0030443	Couples	Konak	2	3	3	3	4	2	4	3	3	3	2	2	07.04.2011	08.53:44
0017843	Family	Alsancak	3	2	3	2	2	2	3	3	4	3	3	2	06.04.2011	14.20:15
0019453	Friends	Alsancak	4	2	3	2	4	3	3	2	4	1	4	2	05.04.2011	11:22:21
0028597	Solo	Basmane	4	2	3	2	4	1	4	3	4	1	4	1	05.04.2011	10.42:41
0033986	Couples	Konak	3	1	4	2	3	3	4	2	2	2	4	2	04.04.2011	22:10:23
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

User Id: User Id is a number given by the rating system for the voters. Each voter gives his/her personal information and enrolls the system. Each time he/she logs on the system this Id is used.

User Type: Each user defines its user type before voting. Because user type is important for the rating assesment. Each user must select one of the five user types:

- Business
- Couples
- Family
- Friends
- Solo

Hotel Id: Each hotel recorded in the assesment system has an Id. Our system uses the name of the hotel as the hotel Id.

Assesment Value: Assesment values are given for the features of the hotel. ‘1’ is the lowest grade and ‘5’ is the highest grade. Assesments are given for the following six contexts, which identify specified hotel features:

- Value of the hotel.
- Rooms of the hotel.
- Location of the hotel.
- Cleanliness of the hotel.
- Service of the hotel.
- Sleep quality of the hotel.

Each assesment value is given together by an importance value I. This value represents the importance of the feature for the voter. Importance value is given in range of 1 to 3. 1 is the lowest and 3 is the highest. If someone does not give an importance value for the feature it is assumed 3.

Time Stamp: Time-stamp is the time of the assesment is completed. Day, month, year, hour and minute of the assesment is recorded.

5.2.4 Processing Raw- Input Data

Raw-Input data is processed by using importance values and processed raw-input data is obtained. Importance values are used in calculations as follows:

- If $I=3$ rating given by the asseser does not change.
- If $I \neq 3$ rating given by the asseser is multipilied by $(1- 0.2/l)$

These importance values are selected for this case as an illustration and can be changed for different cases.

Processed raw-input data example contains following data:

Table 5.2 Processed raw input-data example

User ID	User Type	Hotel ID	Assesment Values						Time Stamp	
			Val	Ro	Lo	Cl	Ser	Sl		
0011183	Business	Konak	4.3	3.4	3.7	4.0	2.3	2.8	12.04.2011	18.00:30
0012185	Business	Alsancak	3.9	3.2	3.9	1.7	2.2	2.7	11.04.2011	15:00:00
0010009	Family	Konak	2.7	3.4	3.3	3.9	3.6	4.0	10.04.2011	16:00:04
0015143	Friends	Konak	2.9	3.5	4.0	3.6	2.8	3.1	09.04.2011	14.00:06
0022145	Solo	Basmane	3.8	3.6	3.9	3.3	4.9	5.1	08.04.2011	18:23:24
0019653	Business	Alsancak	3.2	3.7	3.6	3.3	3.5	3.4	07.04.2011	09.03:05
0030443	Couples	Konak	2.8	3.9	3.7	3.9	3.8	3.1	07.04.2011	08.53:44
0017843	Family	Alsancak	2.9	3.4	3.8	3.8	4.2	3.4	06.04.2011	14.20:15
0019453	Friends	Alsancak	4.4	2.6	4.4	2.9	3.2	3.8	05.04.2011	11:22:21
0028597	Solo	Basmane	3.6	2.8	3.2	4.2	3.2	3.3	05.04.2011	10.42:41
0033986	Couples	Konak	2.4	3.6	3.9	3.6	3.7	3.8	04.04.2011	22:10:23
…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…

5.2.5 Construction of Assesment Matrix

We can represent the relationship between the rater's set U and the rated features of the Hotel Konak in as a *5x6* assesment matrix. Here 5 is the number of elements of the rater's set U and 6 is the number of elements of the hotel features set V. Rows of the matrix represent rater- groups and columns of the matrix represent hotel-features as shown in figure 5.4.

Columns are Hotel Features

$$A_{h.name[t_1,t_2]} = \begin{pmatrix} a_{11} & \cdots & a_{16} \\ \vdots & \ddots & \vdots \\ a_{51} & \cdots & a_{56} \end{pmatrix}$$

Rows are Rater-Groups

Figure 5.4 Trust matrix for the case

Rows of the matrix represent:

$a_{11} - to - a_{16}$: Business Reviewers

$a_{21} - to - a_{26}$: Couples Reviewers

$a_{31} - to - a_{36}$: Family Reviewers

$a_{41} - to - a_{46}$: Friends Reviewers

$a_{51} - to - a_{56}$: Solo Travel Reviewers

Columns of the matrix represent:

$a_{11} - to - a_{51}$: Rater's grade for Value

$a_{12} - to - a_{52}$: Rater's grade for Rooms

$a_{13} - to - a_{53}$: Rater's grade for Location

$a_{14} - to - a_{54}$: Rater's grade for Cleanliness

$a_{15} - to - a_{55}$: Rater's grade for Service

$a_{16} - to - a_{56}$: Rater's grade for Sleep Quality

Assesment matrix A_{5X6} for the Hotel Konak is shown in figure 5.5 by using the processed-raw-input values given in Section 5.2.3. Customers rate the each item in a scale of 1 to 5. 1 is the lowest and 5 is the highest. Arithmetic-mean values are calculated for each rater-group for each-context for the Hotel Konak.

$$A_{konak[t_1,t_2]} = \begin{pmatrix} 4.3 & 3.8 & 3.7 & 4.0 & 2.3 & 2.8 \\ 3.9 & 3.2 & 4.4 & 1.9 & 2.2 & 3.4 \\ 2.7 & 4.1 & 2.8 & 3.1 & 4.1 & 4.6 \\ 3.2 & 2.9 & 2.2 & 3.6 & 3.2 & 4.7 \\ 3.3 & 3.6 & 4.1 & 1.9 & 2.8 & 2.9 \end{pmatrix}$$

Figure 5.5 Assesment matrix for hotel Konak

For example, according to this matrix, rating of family reviewers for the cleanless of the Hotel Konak is : a_{34} =3.1 .

5.2.6 Adding Weights to Assesment Matrix

In section 3.8, we have made assumption-1 about trust value: *Trust in events increases as the number of users tagging the event increases.* Weights of the ratings given by rater-groups are not the the same. The rater-groups tagged more is more valuable for us. The group which gives the the highest number of assesments is the most valuable. We count from the raw the input data and find the values below:

- n_1 : Number of Business Reviewers, 188 assesments.
- n_2 : Number of Couples Reviewers, 156 assesments.
- n_3 : Number of Family Reviewers ,144 assesments.
- n_4 : Number of Friends Reviewers, 123 assesments.
- n_5 : Number of Solo Travel Reviewers, 105 assesments.

Number of assesers are shown in a vector N as follows:

$N_{h.name[t_1,t_2]} = [n_1, n_2, n_3, n_4, n_5, m, time - \text{int}]$ then N_{konak} is as follows:

$$N_{konak[t_1,t_2]} = [188,156,144,123,105,712, time - \text{int}]$$

m is the total number of assesers for the Hotel Konak.

$m = \sum_{i=1}^{5} n_i$ for the Hotel Konak.

If we order the groups according to the the number of assesments they give ordering will be as the following:

n_1, n_2, n_3, n_4, n_5.

n_1 will take the highest weight value and the n_5 will take the least weight value. Definition of trust-weight coefficient is given in Section 3.10.2. Weight-factors can be calculated as follows:

Weight-factor= number of assesments of the rater-group / total number of assesments given by all of the groups. Weight-factors can be calculated as follows:

wf_1 = 188/712= 0.2641

wf_2 = 156/712 = 0.2191

wf_3 = 144/712= 0.2022

wf_4 = 123/712= 0.1727

wf_5 = 101/712 = 0.1419

$$wf_1 + wf_2 + wf_3 + wf_4 + wf_5 = 1$$

Graph between the sets U and V for the context c_1 by considering the weights shown in figure 5.6. This graph is a *weighted* or a *colored graph* since the weights of the edges are added.

Weights of the edges are shown by colors in the graph.

Rank of weights shown by colors from highest to lowest is shown as follows:

- 1. Red
- 2. Orange
- 3. Blue
- 4. Green
- 5. Purple

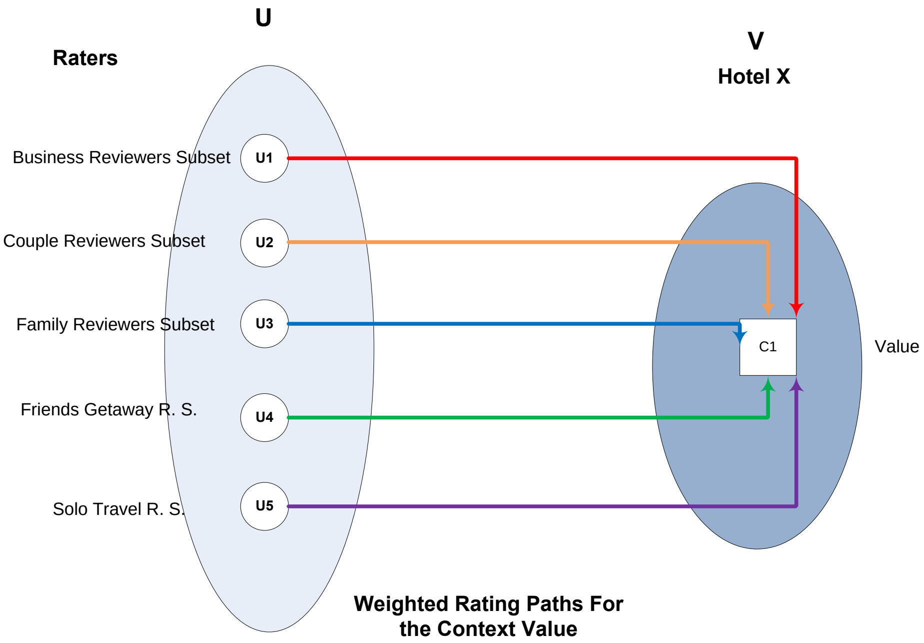


Figure 5.6 Weighted rating edges for the context value

5.2.7 Calculation of the Weighted Assesment Matrix

Our assesment matrix depends on six contexts:

- c_1 (Value for the Hotel): Value means, 'does the hotel deserve the money paid for its all features?' Value is the ratio of money paid/ customer's satisfaction.
- c_2 (Rooms of the Hotel): Rooms of the hotel means, ' how much decoration , comfort and landscape of the room of the hotel satisfy the customer'.
- c_3 (Location of the Hotel): Location of the hotel means, ' does the location of the hotel in the city is suitable for the customer's needs'.

- c_4 (Cleanliness of the Hotel): Cleanliness of the hotel means, ‘how the customer rate cleanless of the room, halls, corridors, restaurants and the lobby of the hotel’.
- c_5 (Service of the Hotel): Service of the hotel means, ‘how the customer rate overall service quality of the the hotel’.
- c_6 (Sleep Quality of the Hotel): Sleep quality of the hotel means, ‘how the customer rate sleep quality of the the hotel’. That means that ‘is the hotel noisy or quiet , is the ambient temperature proper for sleeping, are the beds comfortable or not etc.’

Our assesment-matrix has the following data:

Column 1 is the Rater’s grades for the *context value.*
Column 2 is the Rater’s grade for the *context rooms.*
Column 3 is the Rater’s grade for the *context location.*
Column 4 is the Rater’s grade for the *context cleanless.*
Column 5 is the Rater’s grade for the *context service.*
Column 6 is the Rater’s grade for the *context sleep quality*.

We can easily calculate popularity-values for each context as defined in section 3.9. Since we know the *relative weighted mean* can be expressed by using trust-weight coefficients that sum to one. Such a linear combination is called a *convex combination.*

For value:

$$\left(4.3x0.2641+3.9x0.2191+2.7x0.2022+3.2x0.1727+3.3x0.1419\right)=3.56$$

For rooms:

$$(3.8x0.2641+3.2x0.2191+4.1x0.2022+2.9x0.1727+3.6x0.1419)=3.54$$

For location:

$$\left(3.7x0.2641+4.4x0.2191+2.8x0.2022+2.2x0.1727+4.1x0.1419\right)=3.47$$

For cleanless :

$$\left(4.0x0.2641+1.9x0.2191+3.1x0.2022+3.6x0.1727+1.9x0.1419\right)=3.00$$

For service:

$$\left(2.3x0.2641+2.2x0.2191+4.1x0.2022+3.2x0.1727+2.8x0.1419\right)=2.87$$

For sleep quality:

$$\left(2.8x0.2641+3.4x0.2191+4.6x0.2022+4.7x0.1727+2.9x0.1419\right)=3.64$$

Now, we obtained a assesment-context vector for the time $[t_1, t_2]$interval for the Hotel Konak where t1=01.01.2011, 24.00:00 and t2=30.04.2011, 24:00:00

$$ACV_{konak[t_1,t_2]}=\left[3,56,3,54,3,47,3.00,2.87,3,64,time-\mathrm{int}\right]$$

Time-interval shows the time-gap in which our calculations are made. For our case this the interval *'since the begining of the data began to be collected to the present time'.*

5.2.8 Calculation of the Popularity and Trust Values for Hotels

Our input data contains data of three hotels in the Izmir sub-cluster. These hotels are: Konak, Basmane and Alsancak. Trust-context vectors and assesers vectors for the same time stamp are given as follows:

$$ACV_{konak[t_1,t_2]}=\left[3,56,3,54,3,47,3.00,2.87,3,64,\left[t_1,t_2\right]\right]$$

$N_{konak[t_1,t_2]} = \left[188,156,144,123,105,712,\left[t_1,t_2\right]\right]$

$ACV_{basmane[t_1,t_2]} = \left[4,56,3,94,3,97,4.00,3.87,4,64,\left[t_1,t_2\right]\right]$

$N_{basmane[t_1,t_2]} = \left[392,106,154,133,115,900,\left[t_1,t_2\right]\right]$

$ACV_{alsancak[t_1,t_2]} = \left[4,14,4,04,3,15,3.23,4.07,3,99,\left[t_1,t_2\right]\right]$

$N_{alsancak[t_1,t_2]} = \left[105,101,255,93,99,653,\left[t_1,t_2\right]\right]$

Popularity-index of Hotel Konak in time-interval $\left[t_1,t_2\right]$ for business reviewers can be calculated as follows:

188/(188+392+105) x(3.56+3.54+3.47+3.00+2.87+3.64)=188/685 x 20.08=
0.3038x 20.08=6.10

Popularity-index of Hotel Basmane in time-interval $\left[t_1,t_2\right]$ for business reviwers can be calculated as follows:

392/685 x(4.56+3.94+3.97+4.00+3.87+4.64)= 0.572x24.98=14.29

Popularity-index of Hotel Alsancak in time-interval $\left[t_1,t_2\right]$ for business reviwers can be calculated as follows:

105/682x(4.14+4.04+3.15+3.23+4.07+3.99)=0.1540x22.62=3.48

Most popular Hotel for Business revievers in time-interval $\left[t_1,t_2\right]$ *is Basmane, second Konak and third is Alsancak.*

Most popular Hotel for all groups in time-interval $\left[t_1,t_2\right]$ can be found as follows:

Overall-popularity for the Hotel Konak in time-interval $\left[t_1, t_2\right]$:

712/(712+900+653)x(3.56+3.54+3.47+3.00+2.87+3.64)=712/2265 x 20.08=

0.314x20.08=6.30

Overall-popularity for the Hotel Basmane in time-interval $\left[t_1, t_2\right]$:

900/2265x (4.56+3.94+3.97+4.00+3.87+4.64)=0.397x22.62=8.98

Overall-popularity for the Hotel Alsancak in time-interval $\left[t_1, t_2\right]$:

653/2265x(4.14+4.04+3.15+3.23+4.07+3.99)=0.288x22.62=6.51

Now the ordering is has changed. First popular Hotel in time-interval $\left[t_1, t_2\right]$ is *Basmane but the second is Alsancak and the third is Konak.*

Overall-trust values are calculated by normalizing the previously obtained popularity index values. Popularity index can be 30 highest. Trust values are obtained by dividing the popularity-index values by 30.

Trust-value for the Hotel Konak in time-interval $\left[t_1, t_2\right]$: 6.30/30=0.210

Trust-value for the Hotel Basmane in time-interval $\left[t_1, t_2\right]$: 8.98/30=0.299

Trust-value for the Hotel Alsancak in time-interval $\left[t_1, t_2\right]$: 6.51/30=0.217

5.2.9 Calculation of the Trust Value Intervals with Confidence Probability

According to assumption-3, if number of assesers for each subject-subset U_{ij} for the context V_j $n \geq 30$ in time interval $\left[t_1, t_2\right]$ assesers grades are assumed normally distributed. By using definitions 3.12 and 3.13 confidence interval for a selected confidence probability is calculated as follows:

$$P\left(\mu - z_\alpha \frac{\sigma}{\sqrt{n}} \leq otrust_i \leq \mu + z_\alpha \frac{\sigma}{\sqrt{n}}\right) = \alpha$$

- α (Alpha): Confidence probability
- μ (Arithmetic mean): Arithmetic mean of grades of assesers of the subject-subset U_{ij} for the context V_j $n \geq 30$ in time interval $\left[t_1, t_2\right]$ (grades can be weighted).
- σ (Sigma): Standart deviation of assesers grades.
- n : number of assesers for each subject-subset U_{ij} for the context V_j, n can not be smaller than 30 according to assumption -3.
- z_α : z_α value can be found from Table-3 according to the chosen α value.
- *otrust*: Overall trust value.

otrust value with α probability will lie in the interval :

$$\left(\mu - z_\alpha \frac{\sigma}{\sqrt{n}}, \mu + z_\alpha \frac{\alpha}{\sqrt{n}}\right) \quad \text{where } \mu = otrust_i$$

μ , α , σ values are obtained from the processed-raw input data.

For the example given in section 4.9, if we choose confidence probabilty α =0.90,

$z_\alpha = 1.65$. $otrust_i =$ 0.210 for the Hotel Konak in time-interval $\left[t_1, t_2\right]$, n=712 and standart deviation $\sigma = 1.1$:

$$z_\alpha \frac{\sigma}{\sqrt{n}} = 1.65* \frac{1.1}{\sqrt{712}} = 1.65 * \frac{1.1}{26.68} =1.65 * 0.04123=0.068$$

Since $\mu = otrust_i$, with %90 confidence overalltrust value for Hotel Konak is in the $\left(0.210-0.068, 0.210+0.068\right) = \left(0.142, 0.278\right)$ interval.

For the example given in section 4.9, if we choose confidence probabilty α =0.95, $z_\alpha = 1.96$. $otrust_i =$ 0.210 for the Hotel Konak in time-interval $\left[t_1, t_2\right]$, n=712 and and standart deviation $\sigma = 1.1$:

$$z_\alpha \frac{\sigma}{\sqrt{n}} = 1.96* \frac{1.1}{\sqrt{712}} = 1.96 * \frac{1.1}{26.68} =1.96 * 0.04123=0.081$$

Since $\mu = otrust_i$, with %95 confidence overalltrust value for Hotel Konak is in the $\left(0.210-0.081, 0.210+0.081\right) = \left(0.129, 0.291\right)$ interval.

For the example given in section 4.9, if we choose confidence probabilty α =0.99, $z_\alpha = 2.58$. $otrust_i =$ 0.210 for the Hotel Konak in time-interval $\left[t_1, t_2\right]$, n=712 and and standart deviation $\sigma = 1.1$:

$$z_\alpha \frac{\sigma}{\sqrt{n}} = 2.58* \frac{1.1}{\sqrt{712}} = 2.58 * \frac{1.1}{26.68} =2.58 * 0.04123=0.106$$

Since $\mu = otrust_i$, with %99 confidence overalltrust value for Hotel Konak is in the $\left(0.210-0.106, 0.210+0.106\right) = \left(0.104, 0.316\right)$ interval.

As can be seen from numerical results confidence interval around the mean value becomes larger as confidence increases.

For %90 confidence confidence interval is: 0.278-0.142=0.136

For %95 confidence confidence interval is: 0.291-0.129=0.162

For %99 confidence confidence interval is: 0.316-0.104=0.212

Hotel trust assesmentsystem is a numerical application example of our model. Our contributions are importance value and calculation of total trust as real number intervals in the range of $[0,1]$ by using confidence probability. Our model is so flexible and can be applied to any kind of survey easily. Flexibility of our model is shown in the following case study.

5.3 Turkish Hospital Trust Assesment System

In this scenario we will try to estimate trust relationships for helping decision of patients. Bipartite graphs will be used for modelling. A bipartite graph is a graph where nodes can be divided into two seperate groups *U* and *V* such that no edge connects the vertices in the same group. In our model we have two sets of entities:

U: Subjects (or Raters) who rates the hospital they stayed during their medical treatment. Raters set is composed of four subsets:

- Ambulatory Treatment Patients Subset.
- Inpatients Subset.
- Surgical Treatment Patients Subset.
- Emergency Patients Subset

V : Objects (or Rated-Entities): Hospital X features rated by the patients.

Elements of the set V are real values representing the rater’s grades for the features of the hospital. Features set is composed of six subsets:

- Patient's grade for Cleanliness.
- Patient's grade for Quality of Doctors.
- Patient's grade for Quality of Staff.
- Patient's grade for Concern.
- Patient's grade for Medical Treatment Quality.
- Patient's grade for Treatment Expenditures.

The relationship between these two groups is reflected by the edges.

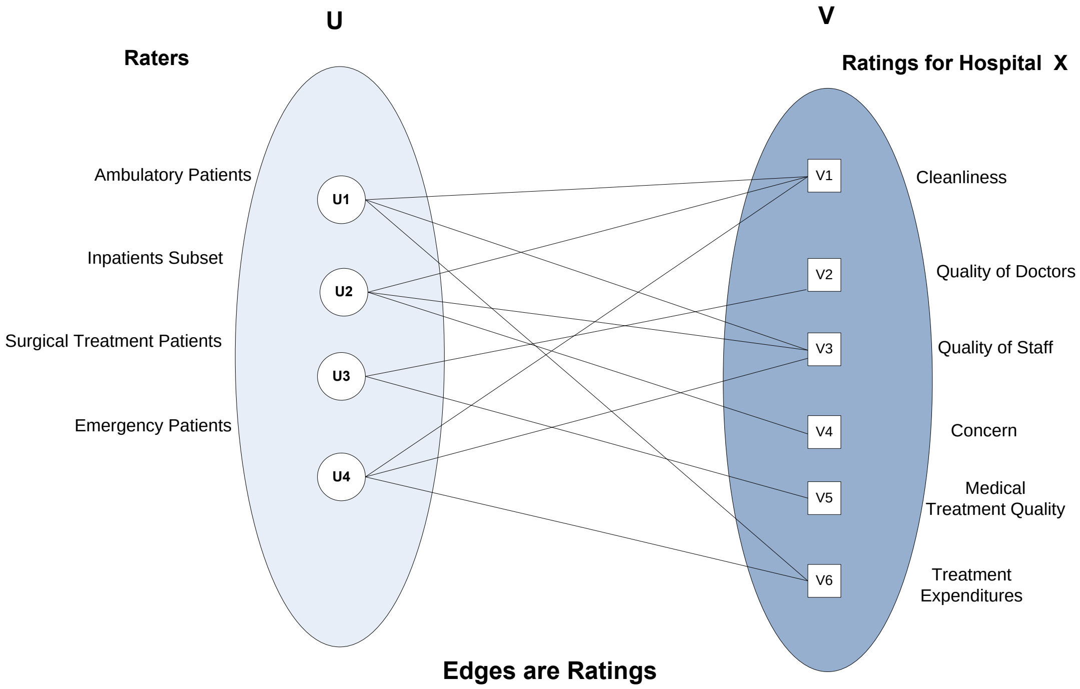


Figure 5.7 Bipartite-graph modelling of hospital trust assesment system

5.3.1 Modelling Hierarchy of Clusters

We begin describing the layout of the model as a bipartite graph of objects and subjects. Both objects and subjects have an hierachy. As shown in Figure 64. Object Set has four subsets. Hospital X is also only a leaf of the Cluster of Subjects.

5.3.2 Tree-like Structure of Clusters

In the hierarchical procedures, we construct a hierarchy or tree-like structure to see the relationship among entities. In our example, entities are clusters of hospitals. Root is the largest cluster containing all hospitals in the Turkey. Sub-clusters are hospitals of cities of Turkey. Izmir hospitals is a sub-cluster of hospitals of Turkey. Hospitals of Izmir are city hospitals like Deu, Ege and Trafik hospitals. Tree-like structure of hospitals of Turkey is shown in figure 5.8.

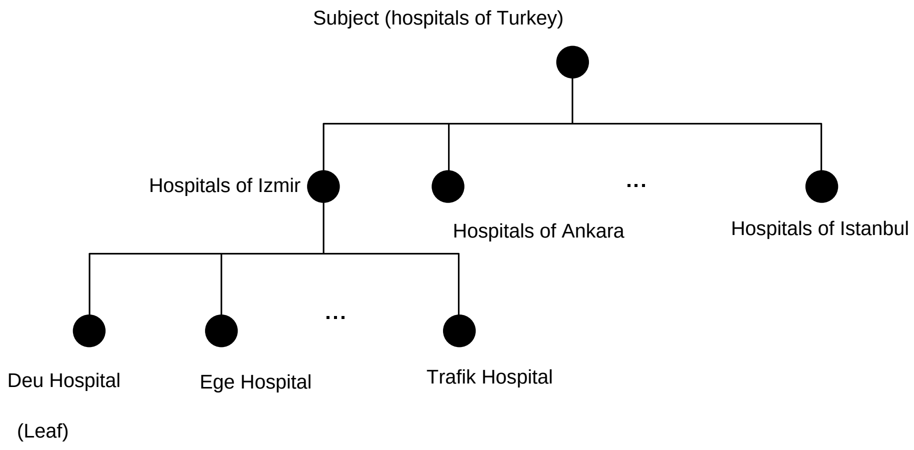


Figure 5.8 Tree like structure of clusters

Tree-like structure of the hospitals of Turkey can be also shown as a Wenn-diagram as shown in figure 5.9.

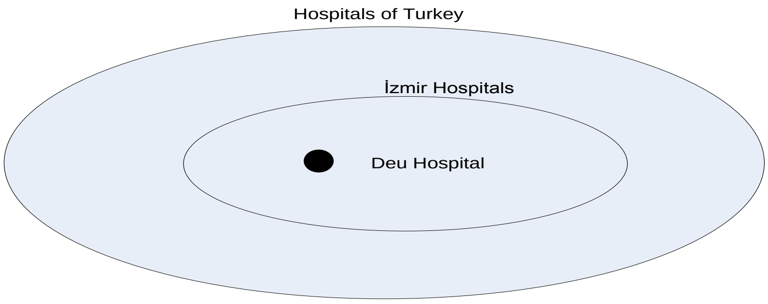


Figure 5.9 Wenn-diagram representation of hierarchy

5.3.3 Raw-Input Data

Hospital trust assesment case study is based on some hypothetical raw-input data for three hospitals in Izmir sub-cluster. Raw input-data example is shown in table 5.3.

Raw- input data example contains following data:

Table 5.3 Raw input-data example

User Id	User Type	Hospital ID	Assesment Values													Time Stamp	
			Cl	I	Doc	I	Sta	I	Con	I	Trt	I	Ex	I		
0011183	Inpatient	Deu	9	2	8	2	8	2	8	2	9	2	9	2	12.04.2011	18.00:30
0012185	Ambulatory	Ege	8	2	6	2	6	2	6	2	6	2	4	1	11.04.2011	15:00:00
0010009	Surgical	Ege	6	2	5	2	6	2	5	2	5	1	5	2	10.04.2011	16:00:04
0015143	Emergency	Deu	8	2	8	1	7	2	8	1	8	2	7	2	09.04.2011	14.00:06
0022145	Emergency	Deu	6	2	8	2	7	2	9	2	8	2	7	2	08.04.2011	18:23:24
0019653	Inpatient	Ege	5	1	7	2	5	2	6	1	5	2	3	2	07.04.2011	09.03:05
0030443	Surgical	Trafik	8	2	6	2	6	2	8	2	6	2	5	2	07.04.2011	08.53:44
0017843	Ambulatory	Ege	4	2	5	2	5	2	5	2	4	2	5	2	06.04.2011	14.20:15
0019453	Inpatient	Trafik	5	2	5	2	5	1	6	2	5	1	5	2	05.04.2011	11:22:21
0028597	Surgical	Deu	6	2	9	2	8	1	8	2	8	1	8	1	05.04.2011	10.42:41
0033986	Emergency	Ege	4	1	5	2	5	1	6	2	5	2	5	2	04.04.2011	22:10:23
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…

User Id: User Id is a number given by the rating system for the voters. Each voter gives his/her personal information and enrolls the system. Each time he/she logs on the system this Id is used.

User Type: Each user defines its user type before voting. Because user type is important for the rating assesment. Each user must select one of the four user types:

- Ambulatory Treatment Patients Subset.
- Inpatients Subset.
- Surgical Treatment Patients Subset.
- Emergency Patients Subset

Hospital Id: Each hospital recorded in the assesment system has an Id. Our system uses the name of the hospital as the hospital Id.

Assesment Value: Assesment values are given for the features of the hotel. ‘1’ is the lowest grade and ‘10’ is the highest grade. Assesments are given for the following six contexts, which identify specified hospital features:

- Patient’s grade for Cleanliness.
- Patient’s grade for Quality of Doctors.
- Patient’s grade for Quality of Staff.
- Patient’s grade for Concern.
- Patient’s grade for Medical Treatment Quality.
- Patient’s grade for Treatment Expenditures.

Each assesment value is given together by an importance value I. This value represents the importance of the feature for the voter. Importance value is given in range of 1 to 2. 1 is the lowest and 2 is the highest. If someone does not give an importance value for the feature it is assumed 2.

Time Stamp: Time-stamp is the time of the assesment is completed. Day, month, year, hour and minute of the assesment is recorded.

5.3.4 Processing Raw Input Data

Raw-Input data is processed by using importance values and processed raw-input data is obtained. Importance values are used in calculations as follows:

- If I=2 rating given by the asseser does not change.
- If I=1 rating given by the asseser is multipilied by 0.8.

These importance values are selected for this case as an illustration and can be changed for different cases.

Processed raw-input data example contains following data:

Table 5.4 Processed raw input-data example

User ID	User Type	Hospital ID	Assesment Values						Time Stamp	
			Cl	Doc	Sta	Con	Trt	Ex		
11183	Inpatient	Deu	9	8	8	8	9	9	12.04.2011	18.00:30
12185	Ambulatory	Ege	8	6	6	6	6	4	11.04.2011	15:00:00
10009	Surgical	Ege	6	5	6	5	4	5	10.04.2011	16:00:04
15143	Emergency	Deu	8	6.4	7	6.4	8	7	09.04.2011	14.00:06
22145	Emergency	Deu	6	8	7	9	8	7	08.04.2011	18:23:24
19653	Inpatient	Ege	4	7	5	4.8	4	3	07.04.2011	09.03:05
30443	Surgical	Trafik	8	6	6	8	6	5	07.04.2011	08.53:44
17843	Ambulatory	Ege	4	5	5	5	4	5	06.04.2011	14.20:15
19453	Inpatient	Trafik	5	5	5	6	5	5	05.04.2011	11:22:21
28597	Surgical	Deu	6	9	6.4	8	6.4	6.4	05.04.2011	10.42:41
33986	Emergency	Ege	3.2	5	4	6	5	5	04.04.2011	22:10:23
…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…

5.3.5 Construction of Assesment Matrix

We can represent the relationship between the rater's set U and the rated features of the Deu Hospital in as a *4x6* assesment matrix. Here 4 is the number of elements of the rater's set U and 6 is the number of elements of the hospital features set V. Rows of the matrix represent rater- groups and columns of the matrix represent hotel-features as shown in figure 5.10.

Columns are Hospital Features

$$A_{hospital.name[t_1,t_2]} = \begin{pmatrix} a_{11} & \cdots & a_{16} \\ \vdots & \ddots & \vdots \\ a_{41} & \cdots & a_{46} \end{pmatrix}$$

Rows are Rater-Groups

Figure 5.10 Trust matrix for the case

Rows of the matrix represent:

$a_{11} - to - a_{16}$: Ambulatory Treatment Patients Reviewers

$a_{21} - to - a_{26}$: Inpatients Reviewers

$a_{31} - to - a_{36}$: Surgical Treatment Patients Reviewers

$a_{41} - to - a_{46}$: Emergency Patients Reviewers

Columns of the matrix represent:

$a_{11} - to - a_{41}$: Rater's grade for Cleanliness.

$a_{12} - to - a_{42}$: Rater's grade for Quality of Doctors.

$a_{13} - to - a_{43}$: Rater's grade for Quality of Staff.

$a_{14} - to - a_{44}$: Rater’s grade for Concern.

$a_{15} - to - a_{45}$: Rater’s grade for Medical Treatment Quality.

$a_{16} - to - a_{46}$: Rater’s grade for Treatment Expenditures.

Assesment matrix A_{4X6} for the Deu hospital Konak is shown in figure 5.11 by using the processed-raw-input values given in Section 5.3.3. Customers rate the each item in a scale of 1 to 10 is the lowest and 10 is the highest. Arithmetic-mean values are calculated for each rater-group for each-context for the Deu hospital.

$$A_{Deu.hospital[t_1,t_2]} = \begin{bmatrix} 8.4 & 7.3 & 6.2 & 6.3 & 7.4 & 8.4 \\ 7.2 & 6.1 & 6.6 & 7.1 & 6.3 & 6.5 \\ 8.1 & 6.6 & 6.8 & 8.2 & 7.0 & 7.1 \\ 8.3 & 7.5 & 7.0 & 8.8 & 7.9 & 5.9 \end{bmatrix}$$

Figure 5.11 Assesment matrix example

For example, according to this matrix, rating of family reviewers for the qulatity of staff of Deu hospital is : a_{34} =8.2.

5.3.6 Adding Weights to Assesment Matrix

In section 3.8, we have made *assumption 3.2* about trust value: *Trust in events increases as the number of users tagging the event increases.* Weights of the ratings given by rater-groups are not the the same. The rater-groups tagged more is more valuable for us. The group which gives the the highest number of assesments is the most valuable. We count from the raw the input data and find the values below:

- n_1 :Number of Ambulatory Treatment Patients Reviewers, 1188 assesments.
- n_2 :Number of Inpatients Reviewers , 702 assesments.
- n_3 :Number of Surgical Treatment Patients, 598 assesments.

- n_4 :Number of Emergency Patients Reviewers, 164 assesments.

Number of assesers are shown in a vector N as follows:

$N_{hospitalname[t_1,t_2]} = [n_1, n_2, n_3, n_4, m, time - \text{int}]$ then $N_{Deu.hospital}$ is as follows:

$$N_{Deu.hospital[t_1,t_2]} = [1188, 702, 598, 164, 2652, time - \text{int}]$$

m is the total number of assesers for the Deu hospital.

$m = \sum_{i=1}^{4} n_i$ for the Deu hospital.

If we order the groups according to the the number of assesments they give ordering will be as the following:

n_1, n_2, n_3, n_4

n_1 will take the highest weight value and the will take the least weight value. Definition of trust-weight coefficient is given in Section 3.9.2. Weight-factors can be calculated as follows:

Weight-factor = number of assesments of the rater-group / total number of assesments given by all of the groups. Weight-factors can be calculated as follows:

wf_1 = 1188/2652= 0.4480

wf_2 = 702/2652 = 0.2647

wf_3 = 598/2652= 0.2255

wf_4 = 164/2652= 0.0618

$wf_1 + wf_2 + wf_3 + wf_4 = 1$

Graph between the sets U and V for the context c_1 by considering the weights shown in figure 5.12. This graph is a *weighted* or a *colored graph* since the weights of the edges are added.

Weights of the edges are shown by colors in the graph.

Rank of weights shown by colors from highest to lowest is shown as follows:

- 1. Red
- 2. Orange
- 3. Blue
- 4. Green

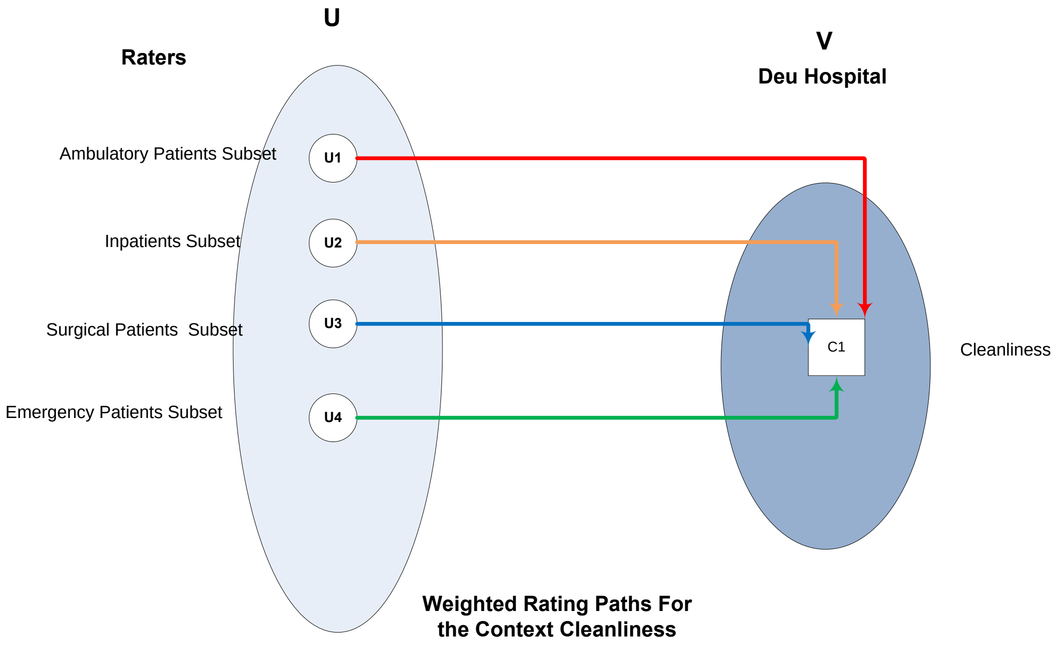

U
Raters
V
Deu Hospital
Ambulatory Patients Subset
U1
Inpatients Subset
U2
Surgical Patients Subset
U3
Emergency Patients Subset
U4
C1
Cleanliness
Weighted Rating Paths For
the Context Cleanliness

Figure 5.12 Weighted rating edges for the context cleanliness

5.3.7 Calculation of the Weighted Assesment Matrix

Our assesment matrix depends on six contexts:

- c_1 (Cleanliness of the hospital): Cleanliness means, ‘does the hospital clean enough?’
- c_2 (Quality of doctors of the hospital): Quality of doctors of the hospital means, ‘ how much doctor’s quality satisfy the patients?’.
- c_3 (Quality of staff of the hospital): Quality of Staff of the hospital means, ‘ does the staff except doctors satisfy the patient’s needs ?’.
- c_4 (Concern of the hospital): Concern of the hospital means, ‘does the patient satisfy from the concern of doctor’s and staff ?’.
- c_5 (Medical treatment quality of the hospital): Medical treatment quality of the hospital means, ‘how the customer rate overall medical treatment quality of the the hospital’.
- c_6 (Treatment expenditures of the hospital): Treatment expenditures of the hospital means, ‘how high the patients medical expenditures in the hospital’.

Our assesment-matrix in figure 5.11 has the following data:

Column 1 is the Rater’s grades for the *context cleanliness.*
Column 2 is the Rater’s grade for the *context quality of doctors.*
Column 3 is the Rater’s grade for the *context quality of staff.*
Column 4 is the Rater’s grade for the *context concern.*
Column 5 is the Rater’s grade for the *context medical treatment quality.*
Column 6 is the Rater’s grade for *the context treatment expenditures.*

We can easily calculate popularity-values for each context as defined in section 3.9. Since we know the *relative weighted mean* can be expressed by using trust-weight coefficients that sum to one. Such a linear combination is called a *convex combination.*

For cleanliness: $(8.4x0.4480+7.2x0.2647+8.1x0.225+8.3x0.0618)=8.01$

For qulity of doctors: $(7.3x0.4480+6.1x0.2647+6.6x0.2255+7.5x0.0618)=6.84$

For quality of staff: $(6.2x0.4480+6.6x0.2647+6.8x0.22255+7.0x0.0618)=6.49$

For concern : $(6.3x0.4480+7.1x0.2647+8.2x0.22255+8.8x0.0618)=7.09$

For medical treatment quality:
$(7.4x0.4480+6.3x0.2647+7.6x0.2255+7.9x0.0618)=7.05$

For treatment expenditures:
$(8.4x0.4480+6.5x0.2647+7.1x0.2255+5.9x0.0618)=7.45$

Now, we obtained a assesment-context vector for the time interval $[t_1,t_2]$ for the

$$ACV_{Deu.hospital[t_1,t_2]}=[8.01,6.84,6.49,7.09,7.05,7.45,time-\text{int}]$$

Time-interval shows the time-gap in which our calculations are made. For our case this the interval *'since the begining of the data began to be collected to the present time'*.

5.3.8 Calculation of the Popularity and Trust Values for Hospitals

Our input data contains data of three hospitals in the Izmir sub-cluster. These hospitals are: Deu, Ege and Trafik.

Trust-context vectors and assesers vectors for the same time stamp are given as follows:

$$ACV_{Deu.hospital[t_1,t_2]} = [8.01, 6.84, 6.49, 7.09, 7.05, 7.45, time - \text{int}]$$

$$N_{Deu.hospital[t_1,t_2]} = [1188, 702, 598, 164, 2652, time - \text{int}]$$

$$ACV_{Ege.hospital[t_1,t_2]} = [6.56, 6.94, 6.97, 5.00, 5.87, 5.64, time - \text{int}]$$

$$N_{Ege.hospital[t_1,t_2]} = [892, 606, 254, 133, 1885, time - \text{int}]$$

$$ACV_{Trafik.hospital[t_1,t_2]} = [7,14, 6,04, 6,75, 6.23, 7.07, 6,99, time - \text{int}]$$

$$N_{Trafik.hospital[t_1,t_2]} = [887, 467, 290, 393, 2007, time - \text{int}]$$

Popularity-value of Deu hospital in time-interval $[t_1, t_2]$ for ambulatory patients reviewers can be calculated as follows:

1188/(1188+892+887) x(8.01+6.84+6.49+7.09+7.05+7.45)=1188/2967 x 42.93= 0.4004x 42.93=17.19

Popularity-index of Ege hospital in time-interval $[t_1, t_2]$ for ambulatory patients reviwers can be calculated as follows:

892/(1188+892+887) x(6.56+6.94+6.97+5.00+5.87+5.64)= 0.3006x36.98=11.12

Popularity-index of Trafik hospital time-interval $[t_1, t_2]$ for ambulatory patients reviwers can be calculated as follows:

887/(1188+892+887)x (7.14+6.04+6.75+6.23+7.07+6.99)=0.2990x40.22=12.02

Most popular hospital for ambulatory patients in time-interval $\left[t_1, t_2\right]$ *is Deu, second Trafik and third is Ege.*

Most popular hospital for all groups in time-interval $\left[t_1, t_2\right]$ can be found as follows:

Overall-popularity for the Deu hospital in time-interval $\left[t_1, t_2\right]$:

2652/(2652+1885+2007)x(8.01+6.84+6.49+7.09+7.05+7.45)=2652/6544 x 42.93=

0.4053x42.93=17.40

Overall-popularity for the Ege hospital in time-interval $\left[t_1, t_2\right]$:

1885/(2652+1885+2007) x(6.56+6.94+6.97+5.00+5.87+5.64)=0.2877x36.98=10.61

Overall-popularity for the Trafik hospital in time-interval $\left[t_1, t_2\right]$:

2007/(2652+1885+2007)x (7.14+6.04+6.75+6.23+7.07+6.99)=0.3070x40.22=12.34

The most popular hospital in Izmir in time-interval $\left[t_1, t_2\right]$ is *Deu, the second is Trafikk and the third is Ege.*

Overall-trust values are calculated by normalizing the previously obtained popularity index values. Popularity index can be 60 highest. Trust values are obtained by dividing the popularity-index values by 60.

Trust-value for the Deu hospital in time-interval $\left[t_1, t_2\right]$: 17.40/60=0.290

Trust-value for the Hotel Basmane in time-interval $\left[t_1, t_2\right]$: 10.61/60=0.177

Trust-value for the Hotel Alsancak in time-interval $\left[t_1,t_2\right]$: 12.34/60=0.206

5.3.9 Calculation of the Trust Value Intervals with Confidence Probability

According to *assumption 3.3,* if number of assesers for each subject-subset U_{ij} for the context V_j $n \geq 30$ in time interval $\left[t_1,t_2\right]$ assesers grades are assumed normally distributed. By using *definitions 3.12 and 3.13* confidence interval for a selected confidence probability is calculated as follows:

$$P\left(\mu - z_\alpha \frac{\sigma}{\sqrt{n}} \leq otrust_i \leq \mu + z_\alpha \frac{\sigma}{\sqrt{n}}\right) = \alpha$$

- α (Alpha): Confidence probability
- μ (Arithmetic mean): Arithmetic mean of grades of assesers of the subject-subset U_{ij} for the context V_j $n \geq 30$ in time interval $\left[t_1,t_2\right]$ (grades can be weighted).
- σ (Sigma): Standart deviation of assesers grades.
- n : number of assesers for each subject-subset U_{ij} for the context V_j, n can not be smaller than 30 according to *assumption3.3.*
- z_α : z_α value can be found from Table-3 according to the chosen α value.
- *otrust*: Overall trust value.

otrust value with α probability will lie in the interval :

$$\left(\mu - z_\alpha \frac{\sigma}{\sqrt{n}}, \mu + z_\alpha \frac{\alpha}{\sqrt{n}}\right) \quad \text{where } \mu = otrust_i$$

μ , α , σ values are obtained from the processed-raw input data.

For the example given in section 5.3.8, if we choose confidence probabilty α =0.90, $z_\alpha = 1.65$. $otrust_i = 0.210$ for the Deu hospital in time-interval $\left[t_1, t_2\right]$, n=2652 and for standart deviation $\sigma = 2.7$:

$$z_\alpha \frac{\sigma}{\sqrt{n}} = 1.65 \text{ x} \frac{2.7}{\sqrt{2652}} = 1.65 \text{ x } \frac{2.7}{51.50} = 1.65 \text{ x } 0.05243 = 0.0865$$

Since $\mu = otrust_i$, with %90 confidence overalltrust value for Deu hospital is in the $\left(0.290 - 0.0865, 0.290 + 0.0865\right) = \left(0.2035, 0.3765\right)$ interval.

For the example given in section 5.3.8, if we choose confidence probabilty α =0.95, $z_\alpha = 1.96$. $otrust_i =$ 0.290 for the Deu hospital in time-interval $\left[t_1, t_2\right]$, n=2652 and and standart deviation $\sigma = 2.7$:

$$z_\alpha \frac{\sigma}{\sqrt{n}} = 1.96\text{x} \frac{2.7}{\sqrt{2652}} = 1.96 \text{ x } \frac{2.7}{51.50} = 1.96 \text{ x } 0.05243 = 0.1028$$

Since $\mu = otrust_i$, with %95 confidence overalltrust value for Deu hospital is in the $\left(0.290 - 0.1028, 0.210 + 0.1028\right) = \left(0.1872, 0.3128\right)$ interval.

For the example given in section 5.3.8, if we choose confidence probabilty α =0.99, $z_\alpha = 2.58$. $otrust_i =$ 0.290 for the Deu hospital in time-interval $\left[t_1, t_2\right]$, n=2652 and and standart deviation $\sigma = 2.7$:

$$z_\alpha \frac{\sigma}{\sqrt{n}} = 2.58* \frac{2.7}{\sqrt{2652}} = 2.58 * \frac{2.7}{51.50} = 2.58 * 0.05243 = 0.1353$$

Since $\mu = otrust_i$, with %99 confidence overalltrust value for Hotel Konak is in the $\left(0.290 - 0.1353, 0.290 + 0.1353\right) = \left(0.1547, 0.4253\right)$ interval.

As can be seen from numerical results confidence interval around the mean value becomes larger as confidence increases.

For %90 confidence confidence interval is: 0.3765-0.2035=0.173

For %95 confidence confidence interval is: 0.3128-0.1872=0.126

For %99 confidence confidence interval is: 0.4253-0.1547=0.271

Hospital trust assesment system is a numerical application example of our model. Our contributions are importance value and calculation of total trust as real number intervals in the range of $[0,1]$ by using confidence probability. Flexibility of our model is verified in this case study.

5.4 User's Trust Calculation for e-Government Web Service

So far we have investigated trust propogation for human to human interactions. Web-based environments typically span interactions between humans and software services (Yolum and et al., 2003). There are many cases where web-based services interact with other web-based services automatically. As an example we will consider the web service e-government.

This web service is in automatic interaction with many other goverment web-services and the user may have or not direct interactions and/or recommendations from second stage web-services. We will assume that e-government web-service interacts with four second stage web-services.These web-sertvices are as following:

- SGK web-service,
- Mernis web-service,
- EGM web-service,
- Justice Ministery web-service.

We do not consider how trust formed between user and second stage web-services in the history. Assume that user's trust and confidence values to the second stage web-services in time interval $(-\infty, t]$ are as following:

$$T_{UWS2_1} = u$$

$$T_{UWS2_2} = 0.4454, \quad \theta_{UWS2_2} = 0.8$$

$$T_{UWS2_3} = 0.5123, \quad \theta_{UWS2_3} = 0.9$$

$$T_{UWS2_4} = 0.5999, \quad \theta_{UWS2_4} = 0.9$$

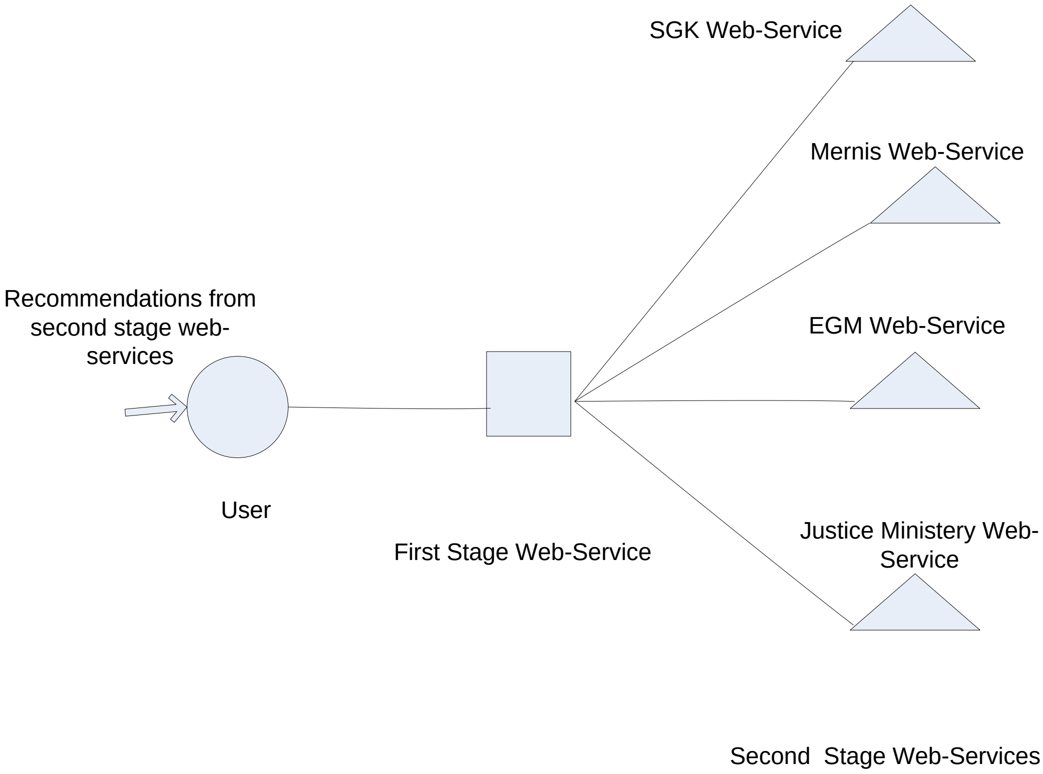

SGK Web-Service
Mernis Web-Service
Recommendations from second stage web-services
EGM Web-Service
User
First Stage Web-Service
Justice Ministery Web-Service
Second Stage Web-Services

Figure 5.13 User interaction with e-government and second stage web-services

Trust between first and second stage web-services also depends on direct interactions between them in history. In this case we consider a service to service interaction.

Trust value formed between web-services in time interval $\left(-\infty, t\right]$ is the number of succesful interactions/number of total interactions between two web-services.

$$T_{WS1-i} = ns_{1-i} / nt_{1-i}$$

$$T_{WS_{1-1}} = ns_{1-1} / nt_{1-1} = 2203/2401 = 0.9175$$

$$T_{WS_{1-2}} = ns_{1-2} / nt_{1-2} = 1003/1456 = 0.6889$$

$$T_{WS_{1-3}} = ns_{1-3} / nt_{1-3} = 2401/2956 = 0.8122$$

$$T_{WS_{1-4}} = ns_{1-4} / nt_{1-4} = 883/956 = 0.9236$$

First stage service to second stage service trust weight factor in time interval $\left(-\infty, t\right]$ is the total number of transactions between service-1 and service-2/ total number of transactions of service-1 with all second stage services.

$$TW_{S1-i} = nt_{1-i} / \sum_{i=1}^{n} nt_{1-i}$$

$$TW_{S1-1} = 2203/7769 = 0.2946$$

$$TW_{S1-2} = 1456/7769 = 0.1876$$

$$TW_{S1-3} = 2956/7769 = 0.3909$$

$$TW_{S1-4} = 956/7769 = 0.1269$$

The problem is to calculate the trust value of the user on the e-government web-service which is in direct interaction between second stage web-services.

A user's trust in time interval $\left(-\infty, t\right]$ to web-service stage-1 which uses the web-service stage-2 is:

T_{UWS1_i} =[$T_{UWS\,2_i}$ + T_{WS1-i}] /2

$$T_{UWS1_1} = 0.9175$$

$$T_{UWS1_2} = \left(0.4454 + 0.6889\right) / 2 = 0.5672$$

$$T_{UWS1_3} = \left(0.5123 + 0.8122\right) / 2 = 0.6623$$

$$T_{UWS1_4} = \left(0.5599 + 0.9236\right) / 2 = 0.7418$$

A user's total trust in time interval $\left(-\infty, t\right]$ to stage-1 web-service which uses *n* web-services of stage-2 is:

T_{UWS1T} = $\sum_{i=1}^{n}$ [T_{UWS1_i} x TW_{S1-i}]

$$T_{UWS1T} = (0.9175x0.2946) + (0.5672x0.1876) + (0.6623x0.3909) + (0.7418x0.1269)$$

$$T_{UWS1T} = 0.2703 + 0.1064 + 0.2589 + 0.0941) = 0.7297$$

Let confidence values for service to service interaction given as following:

$$\theta_{UWS1_1} = 0.9$$

$$\theta_{UWS1_2} = 0.8$$

$$\theta_{UWS1_3} = 0.9$$

$\theta_{UWS1_4} = 0.8$

Confidence values of user in time interval $\left(-\infty, t\right]$ to e-government web-service which uses four web-service stage-2 can be calculated as following:

θ_{UWS1_i} =[θ_{UWS2_i} + θ_{WS1-i}] /2

$\theta_{UWS1_1} = 0.9$

$\theta_{UWS1_2} = (0.8 + 0.8)/2 = 0.8$

$\theta_{UWS1_3} = (0.9 + 0.9)/2 = 0.9$

$\theta_{UWS1_4} = (0.9 + 0.9)/2 = 0.9$

Total confidence of user in time interval $\left(-\infty, t\right]$ to e-government web-service which uses four web-services of stage-2 can be calculated as following:

$$\theta_{UWS1T} = \sum_{i=1}^{n} [T_{UWS1_i} \text{ x } \theta_{UWS1_i}]$$

$\theta_{UWS1T} = (0.9x0.2946) + (0.8x0.1876) + (0.9x0.3909) + (0.9x0.1269)$

$\theta_{UWS1T} = 0.2651 + 0.1501 + 0.3518 + 0.1142 = 0.8812$

Our model calculates the trust value between first and second level services on the success of automated transactions. A second level service with a high number of successful transactions is considered more trustworthy. The main contribution of our model is to consider confidence value propogation between services. In this case study a numerical example of confidence propogation is given. Confidence propogation is completely neglected in previoes similar researches.

CHAPTER SIX

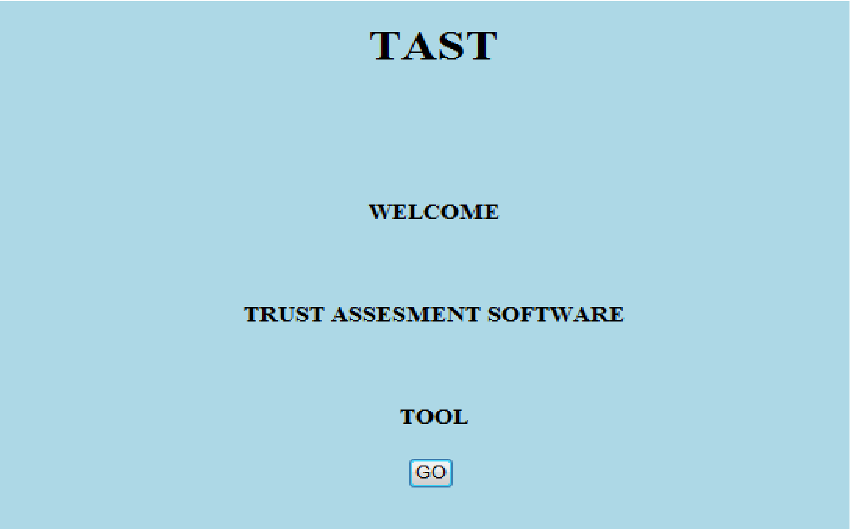
TRUST ASSESMENT SOFTWARE TOOL (TAST)

6.1 Motivation

Management of obtained information from web-based surveys is a very important task for organizations. Assesment of results should designate the organization's target customer base, the weaknesses and strengths of their services . Results should be benefical for designing new marketing strategies. Our software aims to be applicable for one organization or a group of organizations having operations on the same business field. The software should be able to calculate trust values in selected time intervals. This feature should make it possible to observe and compare system trust value changes of an organization by itself and by its competitors.

6.2 Structure of the TAST Software

TAST software is based on a web application in an object oriented programming language such as PHP and MySQL environment. TAST web-service is reached by clicking the link http://web.deu.edu.tr/anket/. Welcome page is shown in the figure 6.1.



TAST

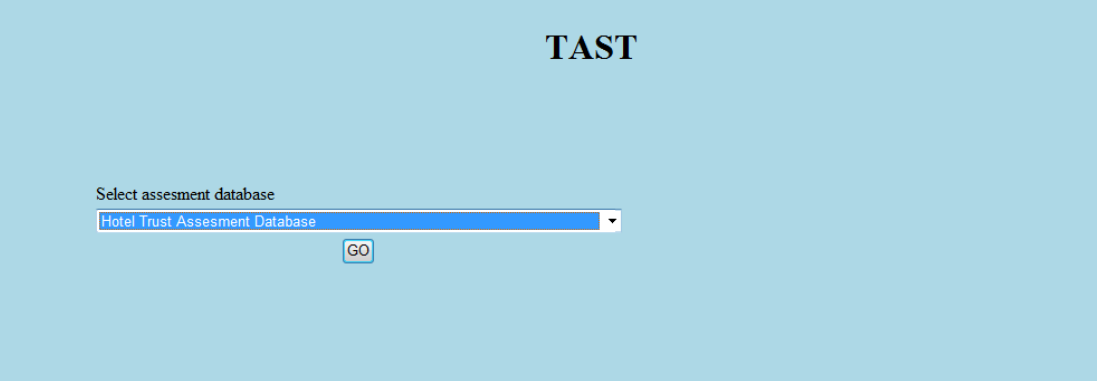
WELCOME

TRUST ASSESMENT SOFTWARE

TOOL

Figure 6.1 Tast welcome page

By clicking *go* button select assessment database page is displayed as shown in figure 6.2.



Select assesment database
Hotel Trust Assesment Database

Figure 6.2 Assesment database selection page

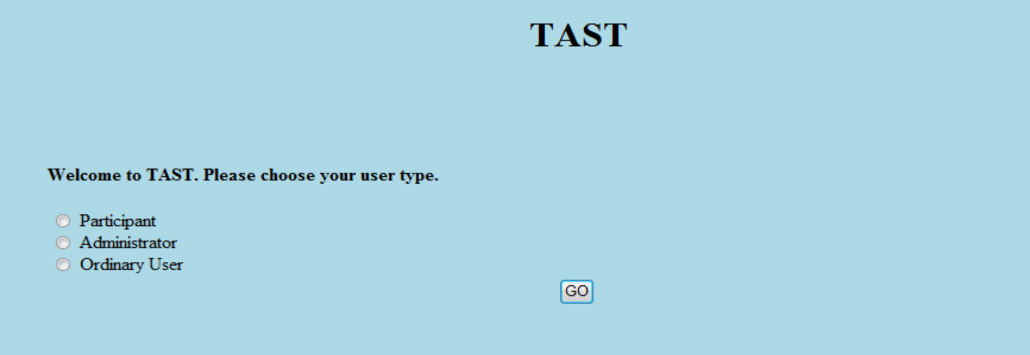
Other database selection options provided by TAST is as following:

- Turkish hospital trust assement database.
- Retail shopping center assesment database.
- Turkish banking system commercial customers trust assesment database.
- Turkish banking system personal customers trust assesment database.

Number of databases can be increased by demand. Only hotel trust assesment database is activated by hypothetical data for testing the software.

6.2.1 User Type Selection

When the user select one of the databases and clicks the *go* buton user type selection page is displayed as shown in figure 6.3.



Welcome to TAST. Please choose your user type.

○ Participant
○ Administrator
○ Ordinary User

Figure 6.3 User type selection page

Three user types which have different user types are defined:

- Participants.
- Administrators.
- Ordinary users.

Access rights of user types are defined by the UML use case diagram in figure 6.4.

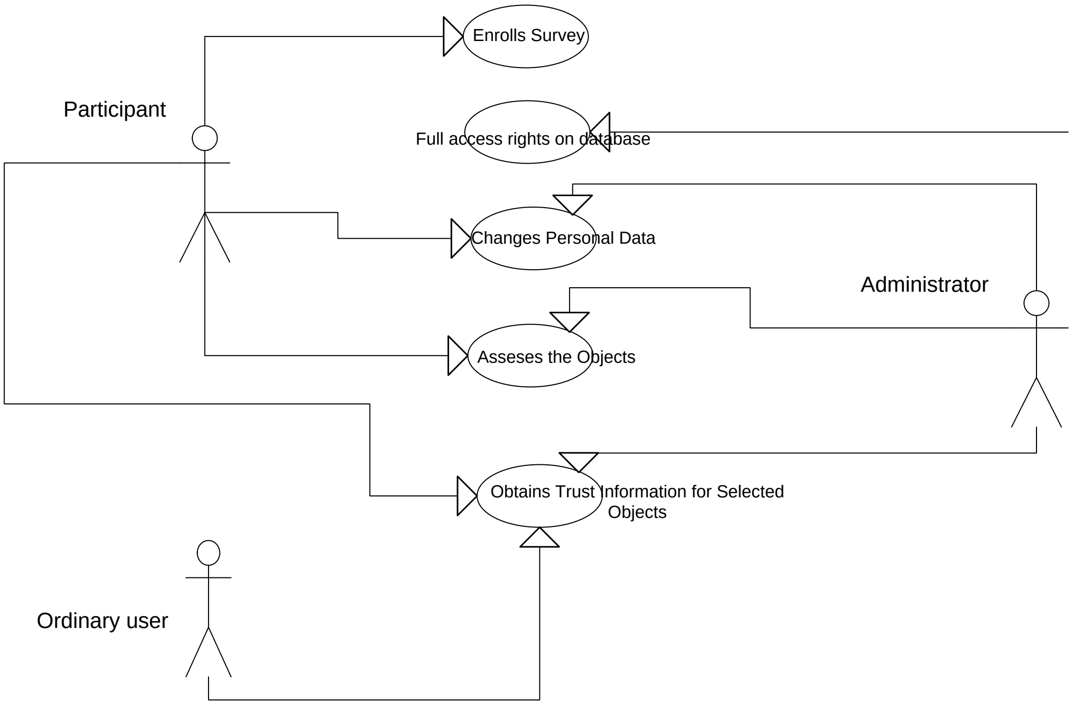


Figure 6.4 UML use case diagram of user access rights

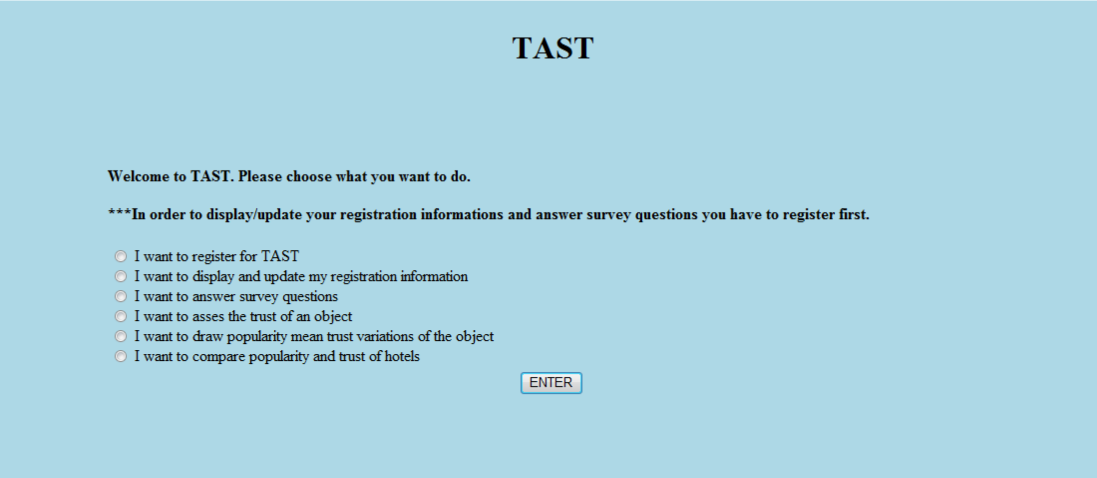
Ordinary users have least access rights. They can oly select and view the trust information of selected objects.

Administrators are password protected and have full access rights on the TAST software. Maximum three administrators can be defined for the TAST.

Participants can enroll the survey and can answer the survey questions. They can assess the trust of selected objects.

6.2.2 Survey Enrollment Procedure

To enroll the survey as a participant select the participant option as shown in figure 6.3 and click *go* button. The page shown in figure 6.5 is displayed.



TAST

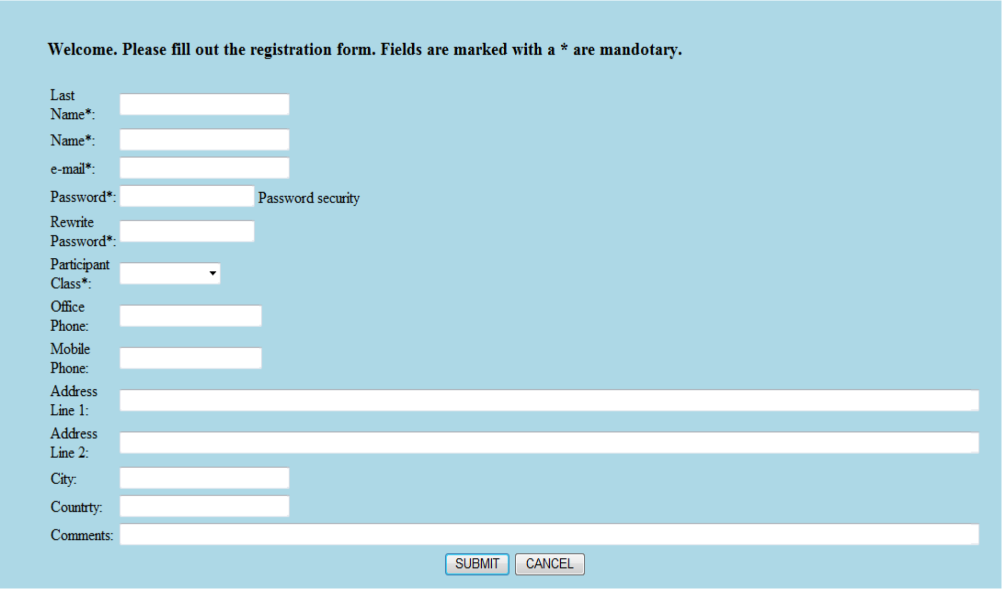
Welcome to TAST. Please choose what you want to do.

*****In order to display/update your registration informations and answer survey questions you have to register first.**

○ I want to register for TAST
○ I want to display and update my registration information
○ I want to answer survey questions
○ I want to asses the trust of an object
○ I want to draw popularity mean trust variations of the object
○ I want to compare popularity and trust of hotels

Figure 6.5 Participant options selection page

Participants enroll the survey by choosing the first option and by clicking enter. Participants must fill the registration page by giving some personal information as shown in figure 6.6.



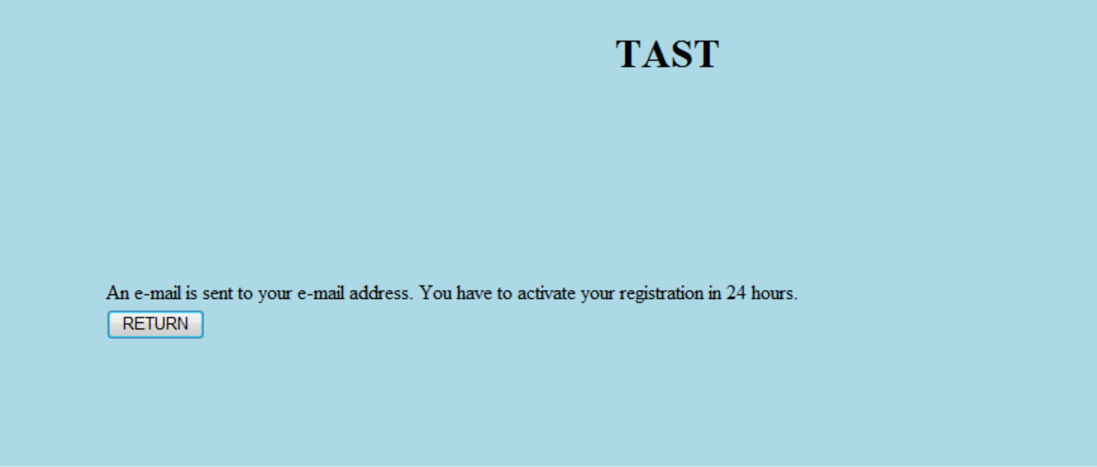
Welcome. Please fill out the registration form. Fields are marked with a * are mandotary.

Last Name*:
Name*:
e-mail*:
Password*: Password security
Rewrite Password*:
Participant Class*:
Office Phone:
Mobile Phone:
Address Line 1:
Address Line 2:
City:
Countrty:
Comments:

Figure 6.6 Participant registration page

Participant must give a strong password otherwise registration process can not be completed.

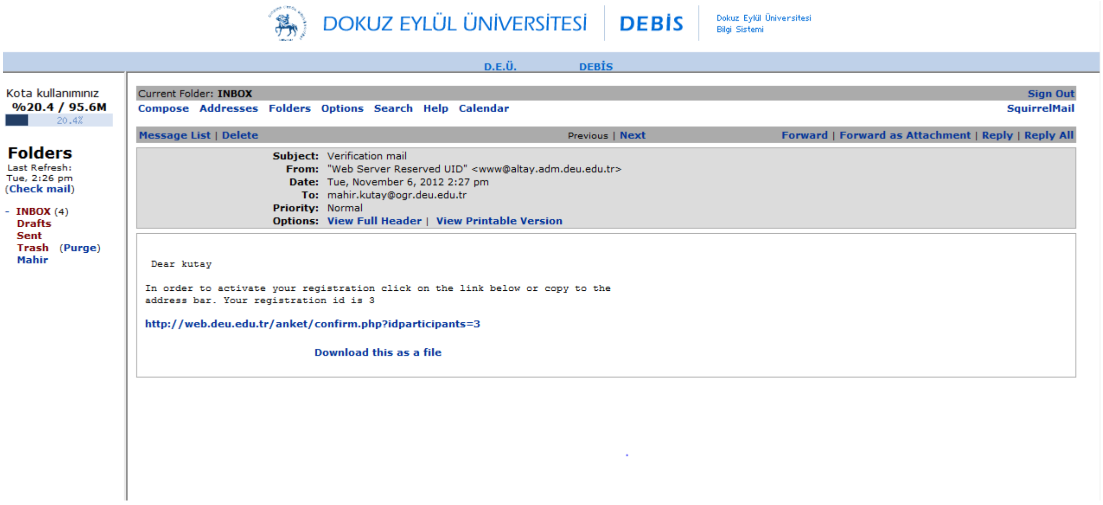
Participant fills the blank parts of the page and clicks the *enter* button. A page notifying an e-mail sent to participant is displayed as shown in figure 6.7. E-mail contains registration Id and an activation link which expires in twenty-four hours.



TAST

An e-mail is sent to your e-mail address. You have to activate your registration in 24 hours.

Figure 6.7 E-mail notification page



Current Folder: **INBOX**

Subject: Verification mail
From: "Web Server Reserved UID" <www@altay.adm.deu.edu.tr>
Date: Tue, November 6, 2012 2:27 pm
To: mahir.kutay@ogr.deu.edu.tr
Priority: Normal
Options: View Full Header | View Printable Version

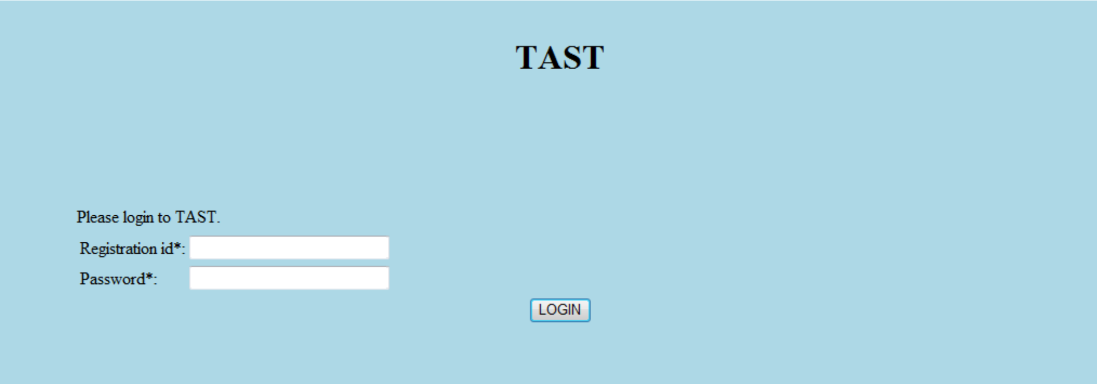
```
 Dear kutay

In order to activate your registration click on the link below or copy to the
address bar. Your registration id is 3
```

http://web.deu.edu.tr/anket/confirm.php?idparticipants=3

Figure 6.8 E-mail with activation link

Participant can change own personal information by selecting the second option in figure 6.5 . This time a page requesting participant's registration Id and password is displayed. This page is shown in figure 6.9.



TAST

Please login to TAST.

Registration id*:

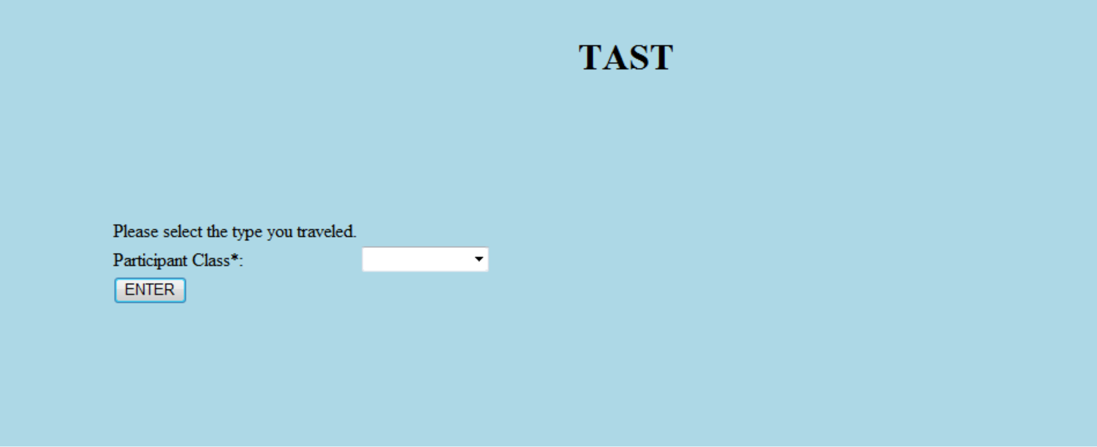
Password*:

Figure 6.9 Participant login page

6.2.3 *Answering Survey Questions*

Participant can answer survey questions by selecting the third option in figure 6.5. Participant should give his/her registration Id and password on the page displayed as shown in in figure 6.9.

When a participant logs in the survey a page for entering the participant class is displayed as shown in figure 6.10.



TAST

Please select the type you traveled.

Participant Class*:

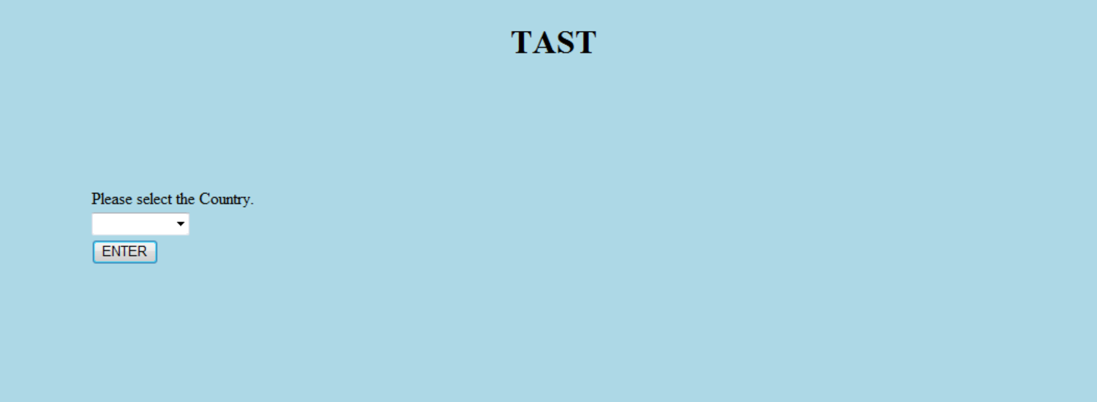
Figure 6.10 Participant class selection page

Participant must select one of the participant classes given in the drop down menu.. For the hotel trust assesment system participant classes is given as following:

- Businessman
- Couple

- Family
- Friend group
- Solo traveler

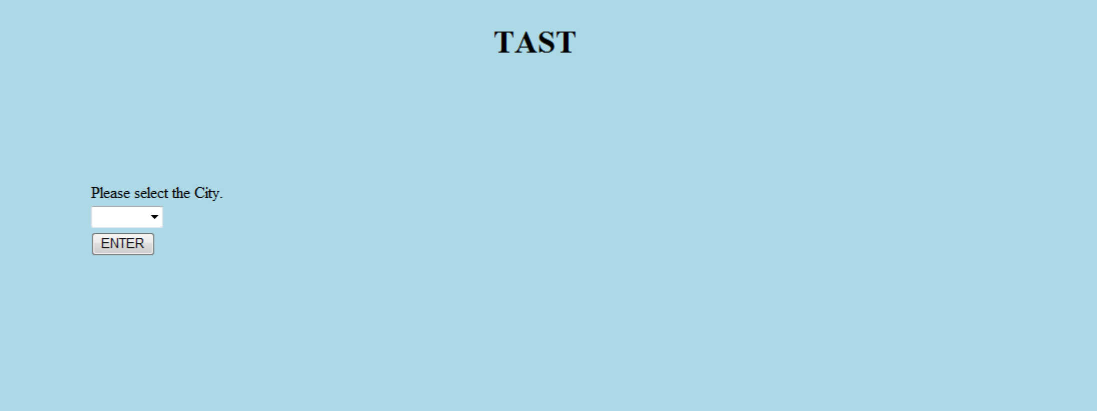
When selection is completed and enter button clicked country selection page is displayed as shown in figure 6.11


TAST

Please select the Country.

Figure 6.11 Country of the hotel selection page

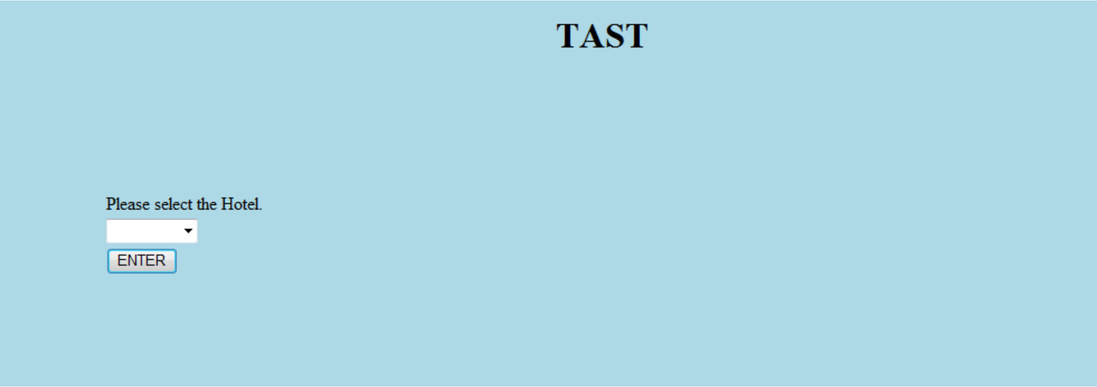
Participant should select the country of the hotel by using the drop-down menu and click enter. City of the hotel selection page is displayed as shown in figure 6.12.


TAST

Please select the City.

Figure 6.12 City of the hotel selection page

When selection is completed and enter button clicked hotel selection page is displayed as shown in figure 6.13.

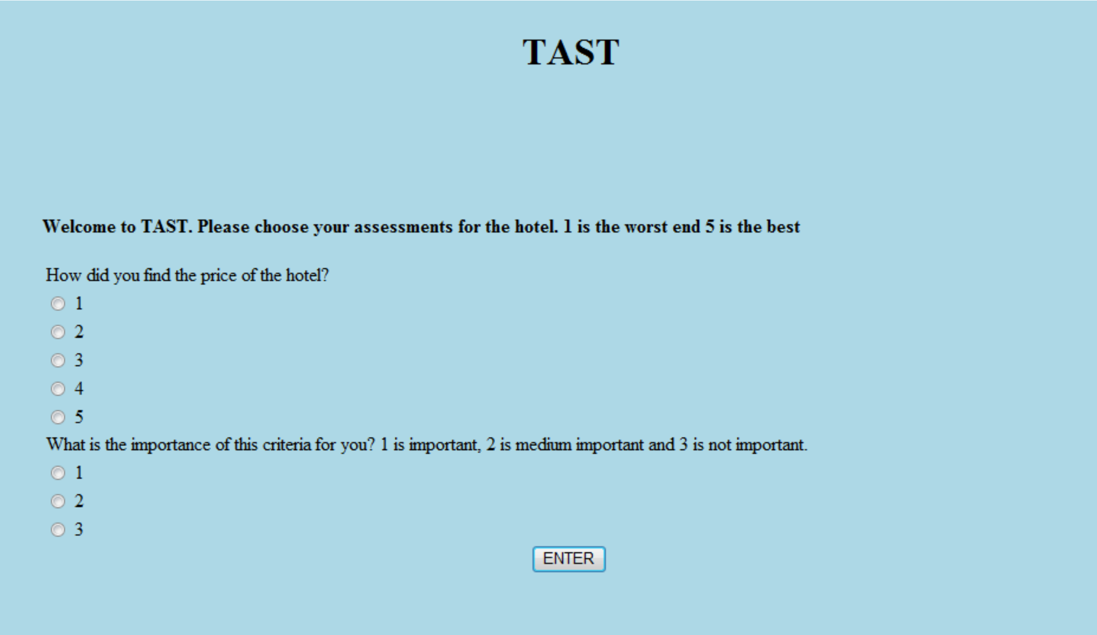


TAST

Please select the Hotel.

Figure 6.13 Hotel selection page

Participant should select the name of the hotel by using the drop-down menu and click enter. First two questions of the survey about the price of the hotel and its importance is displayed as shown in figure 6.14.



TAST

Welcome to TAST. Please choose your assessments for the hotel. 1 is the worst end 5 is the best

How did you find the price of the hotel?

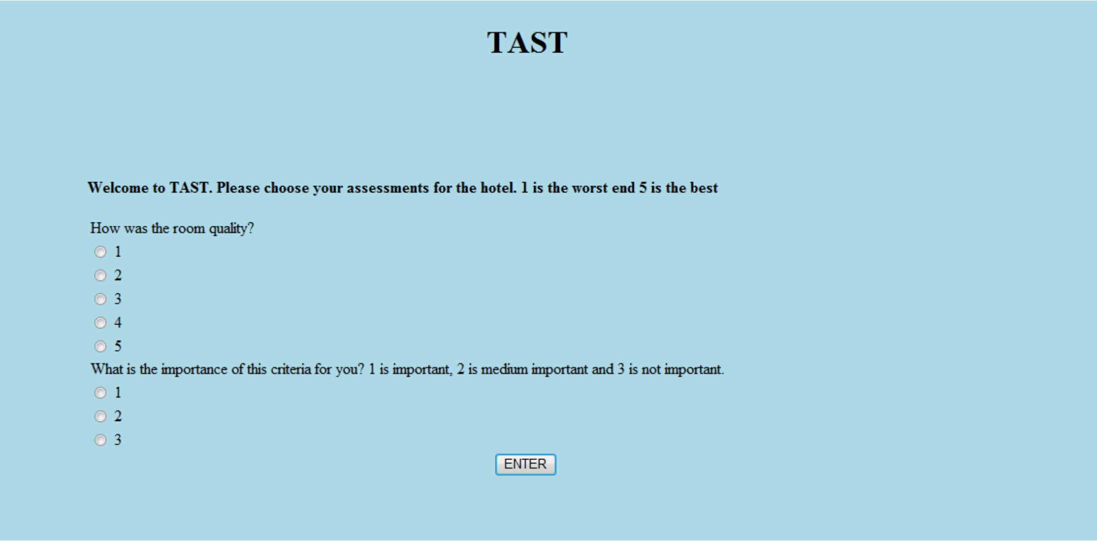
○ 1
○ 2
○ 3
○ 4
○ 5

What is the importance of this criteria for you? 1 is important, 2 is medium important and 3 is not important.

○ 1
○ 2
○ 3

Figure 6.14 Survey questions about the price of the hotel and its importance for the participant

Participant choose the assessment values by clicking only one of the grades for the price of the hotel and its importance for the participant. Two selections at the same time or no selection for each question is not allowed. By clicking enter the next page is displayed as shown in figure 6.15.



TAST

Welcome to TAST. Please choose your assessments for the hotel. 1 is the worst end 5 is the best

How was the room quality?

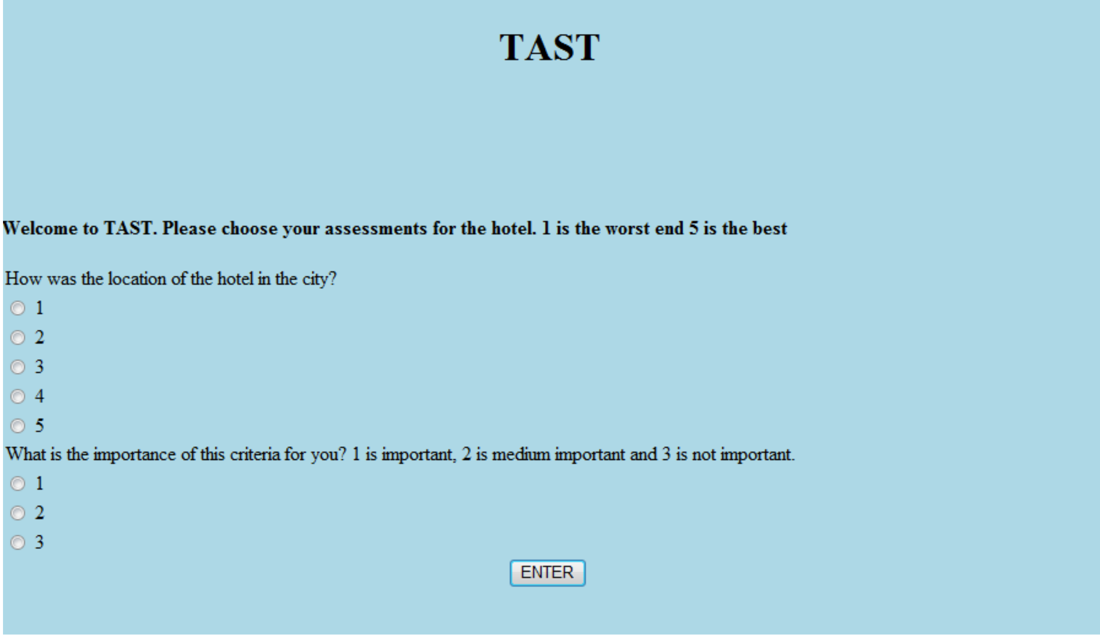
○ 1
○ 2
○ 3
○ 4
○ 5

What is the importance of this criteria for you? 1 is important, 2 is medium important and 3 is not important.

○ 1
○ 2
○ 3

Figure 6.15 Survey questions about the room quality of the hotel and its importance

Participant choose the assessment values by clicking only one of the grades for the room quality of the hotel and its importance for the participant. Two selections at the same time or no selection for each question is not allowed. By clicking enter the next page is displayed as shown in figure 6.16.



TAST

Welcome to TAST. Please choose your assessments for the hotel. 1 is the worst end 5 is the best

How was the location of the hotel in the city?

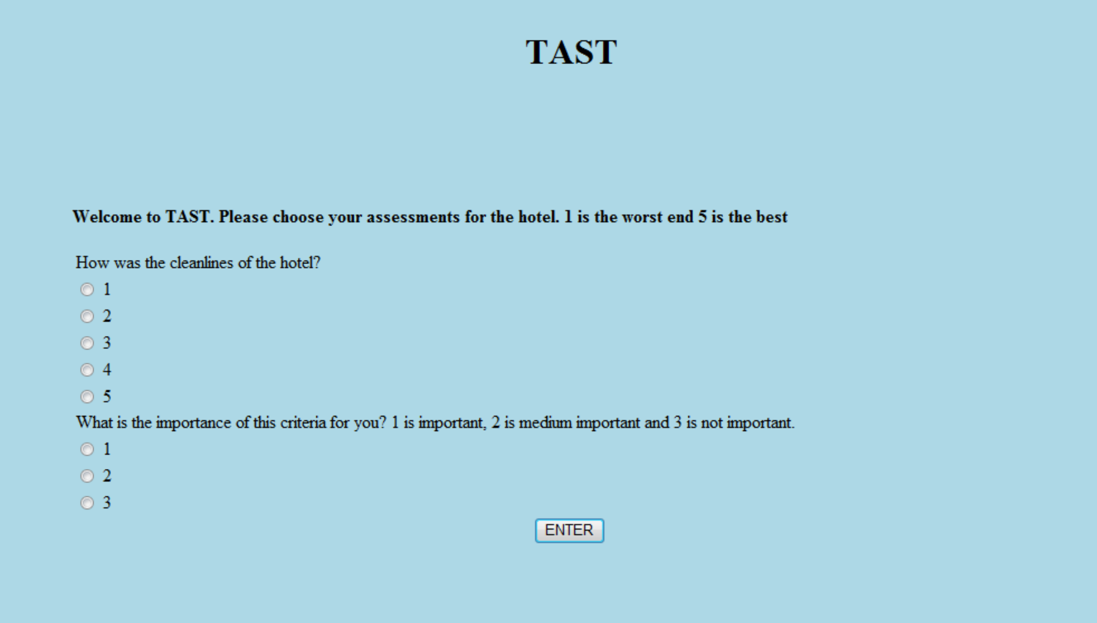
○ 1
○ 2
○ 3
○ 4
○ 5

What is the importance of this criteria for you? 1 is important, 2 is medium important and 3 is not important.

○ 1
○ 2
○ 3

Figure 6.16 Survey questions about the location of the hotel and its importance for the participant

Participant choose the assessment values by clicking only one of the grades for the location of the hotel and its importance for the participant. Two selections at the same time or no selection for each question is not allowed. By clicking enter the next page is displayed as shown in figure 6.17.



TAST

Welcome to TAST. Please choose your assessments for the hotel. 1 is the worst end 5 is the best

How was the cleanlines of the hotel?

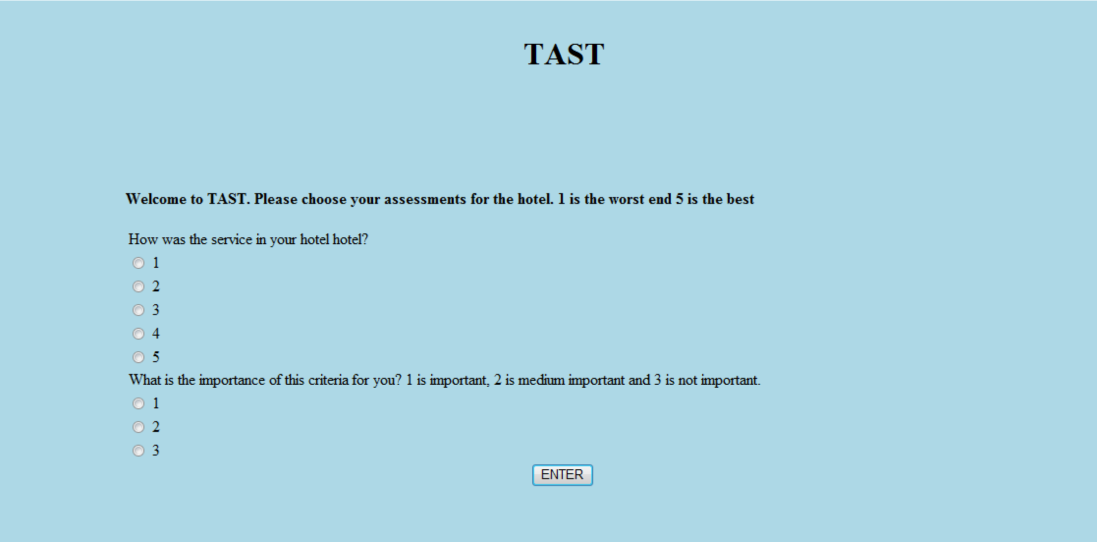
○ 1
○ 2
○ 3
○ 4
○ 5

What is the importance of this criteria for you? 1 is important, 2 is medium important and 3 is not important.

○ 1
○ 2
○ 3

Figure 6.17 Survey questions about the cleanliness of the hotel and its importance for the participant

Participant choose the assessment values by clicking only one of the grades for the cleanliness of the hotel and its importance for the participant. Two selections at the same time or no selection for each question is not allowed. By clicking *enter* the next page is displayed as shown in figure 6.18.



TAST

Welcome to TAST. Please choose your assessments for the hotel. 1 is the worst end 5 is the best

How was the service in your hotel hotel?

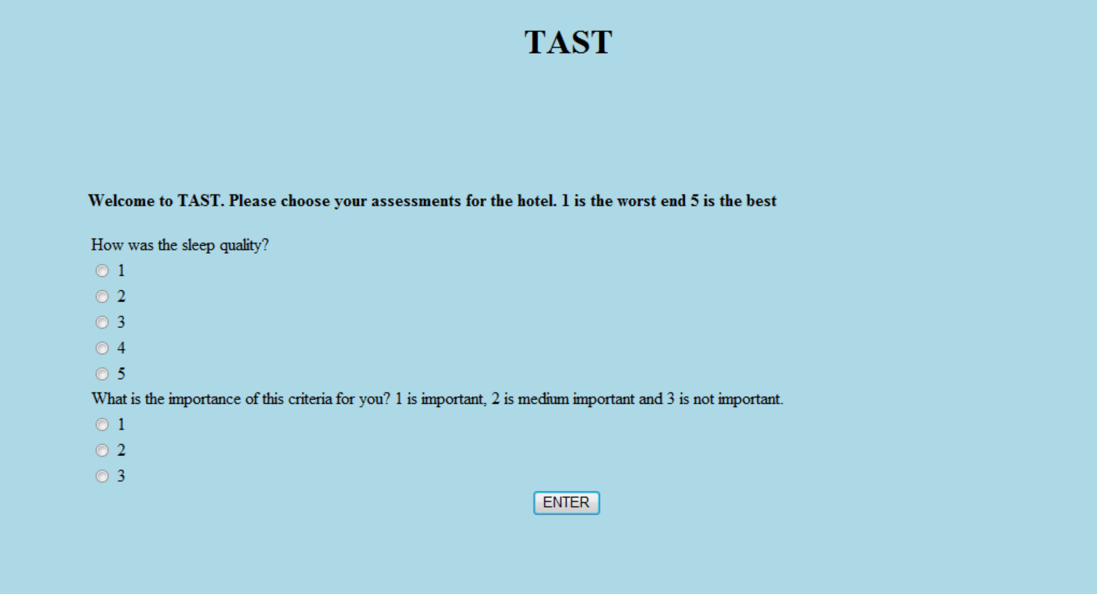
○ 1
○ 2
○ 3
○ 4
○ 5

What is the importance of this criteria for you? 1 is important, 2 is medium important and 3 is not important.

○ 1
○ 2
○ 3

Figure 6.18 Survey questions about the service of the hotel and its importance for the participant

Participant choose the assessment values by clicking only one of the grades for the service of the hotel and its importance for the participant. Two selections at the same time or no selection for each question is not allowed. By clicking *enter* the next page is displayed as shown in figure 6.19.



TAST

Welcome to TAST. Please choose your assessments for the hotel. 1 is the worst end 5 is the best

How was the sleep quality?

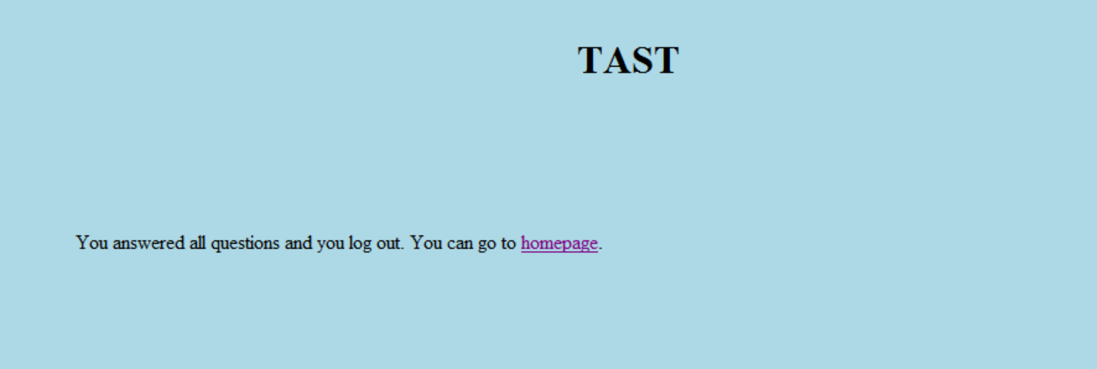
○ 1
○ 2
○ 3
○ 4
○ 5

What is the importance of this criteria for you? 1 is important, 2 is medium important and 3 is not important.

○ 1
○ 2
○ 3

Figure 6.19 Survey questions about the sleep quality of the hotel and its importance

Participant choose the assessment values by clicking only one of the grades for the service of the hotel and its importance for the participant. Two selections at the same time or no selection for each question is not allowed. By clicking *enter* the next page notifying that the survey is ended is displayed as shown in figure 6.20.



TAST

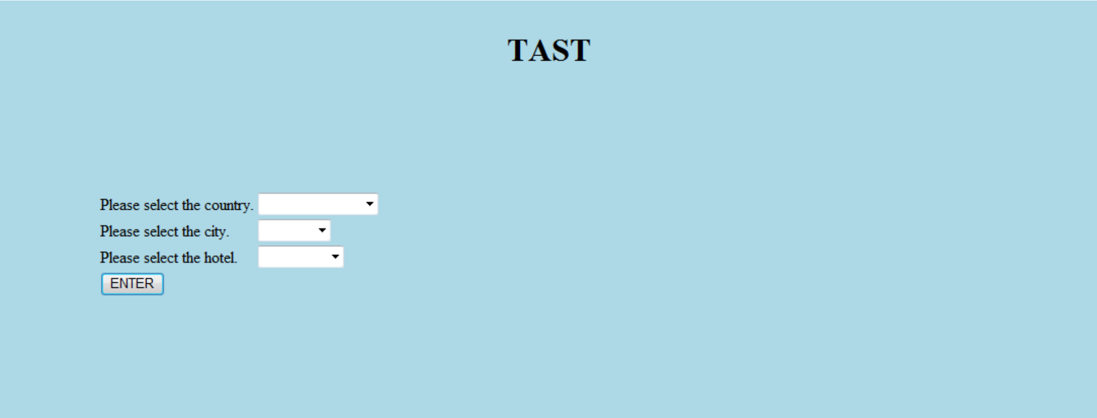
You answered all questions and you log out. You can go to homepage.

Figure 6.20 Page notifying that the survey is ended

Participant can participate the survey with the same Id seventy-two hours later. This feature prohibits successive rating of malicious participants.

6.2.4 Assesing Trust of an Object

To assess the trust of an object the participant should select the fourth option as shown in figure 6.5 and clicking *go* button. The page shown in figure 6.21 is displayed.



TAST

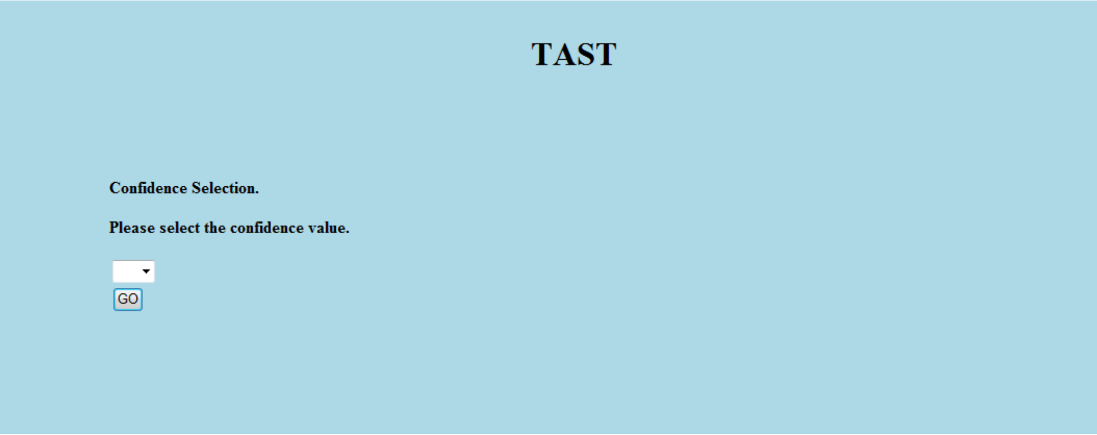
Please select the country.

Please select the city.

Please select the hotel.

Figure 6.21 Page for the selection of object for trust assesment

This page does not require login password. Participant chooses the country, city and the name of the hotel by using drop down menus. When selection completed *enter* button is clicked and the page shown in figure 6.22 is displayed.



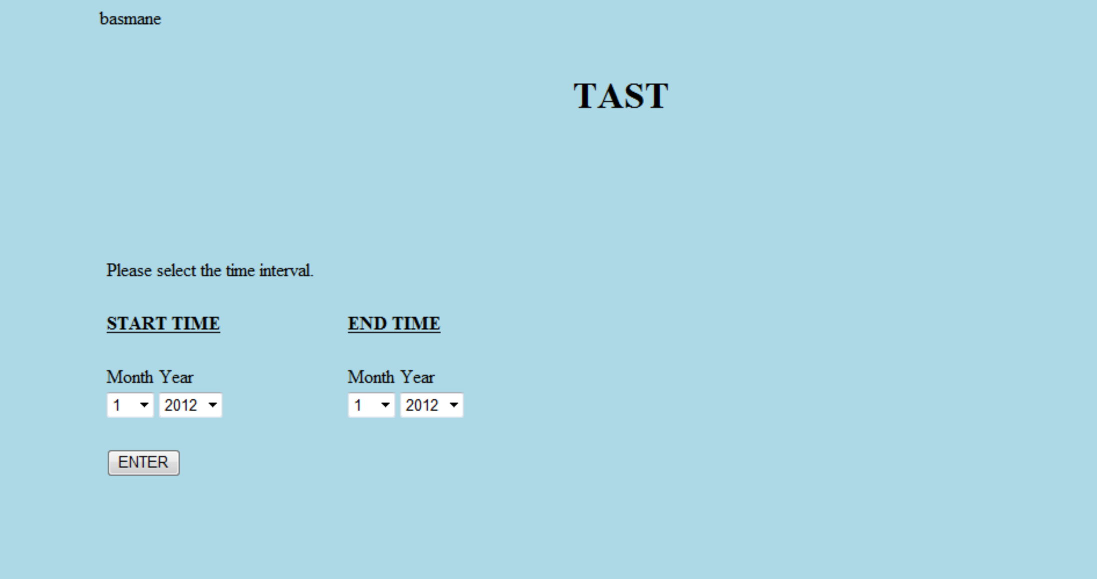
TAST

Confidence Selection.

Please select the confidence value.

Figure 6.22 Page for the selection of confidence value

Confidence value selection can be done in the interval 0.80 to 0.99. Our model assumes useless the values below 0.80. When selection is completed and *go* button clicked, page shown in figure 6.23 is displayed.



basmane

TAST

Please select the time interval.

START TIME

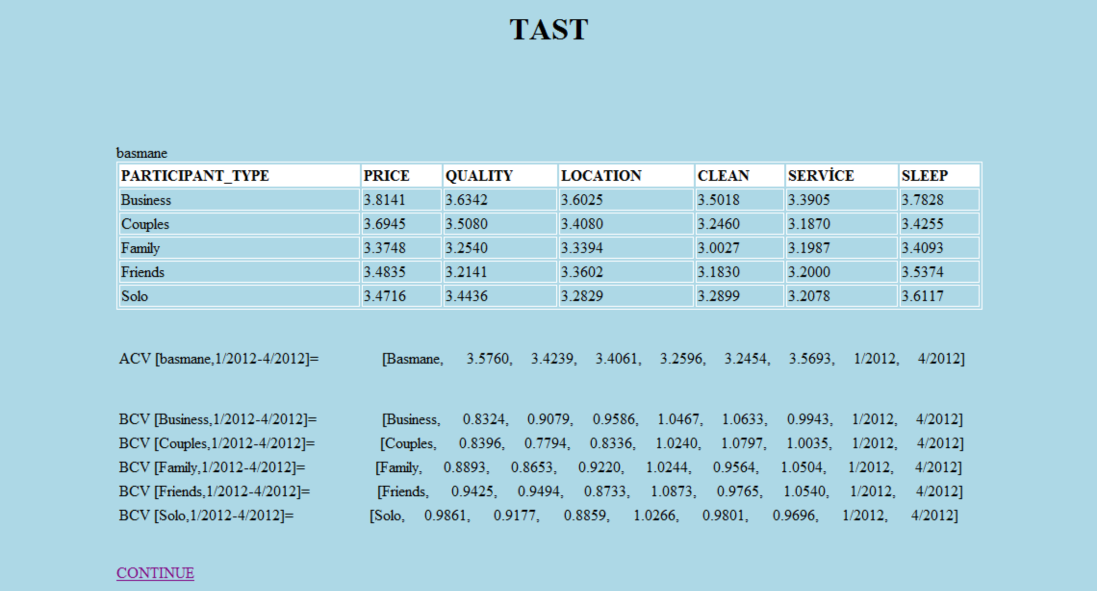
Month Year
1 2012

END TIME

Month Year
1 2012

Figure 6.23 Page for the selection of time interval

Time interval selection can be done in months. Minimum selectable time interval is one month of the year. Upper limit is not defined and depends on the the data collection interval. As an example, let us choose the first four months of the year 2012 and click *enter* button. Page shown in figure 6.24 is displayed.



TAST

basmane

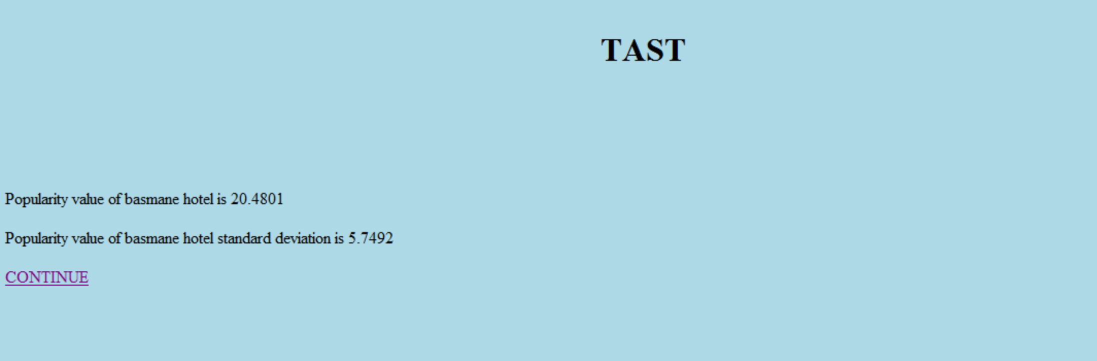
PARTICIPANT_TYPE	PRICE	QUALITY	LOCATION	CLEAN	SERVİCE	SLEEP
Business	3.8141	3.6342	3.6025	3.5018	3.3905	3.7828
Couples	3.6945	3.5080	3.4080	3.2460	3.1870	3.4255
Family	3.3748	3.2540	3.3394	3.0027	3.1987	3.4093
Friends	3.4835	3.2141	3.3602	3.1830	3.2000	3.5374
Solo	3.4716	3.4436	3.2829	3.2899	3.2078	3.6117

```
ACV [basmane,1/2012-4/2012]=      [Basmane,   3.5760,   3.4239,   3.4061,   3.2596,   3.2454,   3.5693,   1/2012,   4/2012]


BCV [Business,1/2012-4/2012]=     [Business,   0.8324,   0.9079,   0.9586,   1.0467,   1.0633,   0.9943,   1/2012,   4/2012]
BCV [Couples,1/2012-4/2012]=      [Couples,   0.8396,   0.7794,   0.8336,   1.0240,   1.0797,   1.0035,   1/2012,   4/2012]
BCV [Family,1/2012-4/2012]=      [Family,   0.8893,   0.8653,   0.9220,   1.0244,   0.9564,   1.0504,   1/2012,   4/2012]
BCV [Friends,1/2012-4/2012]=      [Friends,   0.9425,   0.9494,   0.8733,   1.0873,   0.9765,   1.0540,   1/2012,   4/2012]
BCV [Solo,1/2012-4/2012]=        [Solo,   0.9861,   0.9177,   0.8859,   1.0266,   0.9801,   0.9696,   1/2012,   4/2012]
```

Figure 6.24   Calculated assesment matrix, ACV and BCV vectors

On this page calculated assesment matrix , ACV and BCV vectors of the object for the selected time interval is displayed.When *continue* button clicked page shown in figure 6.25 is displayed.



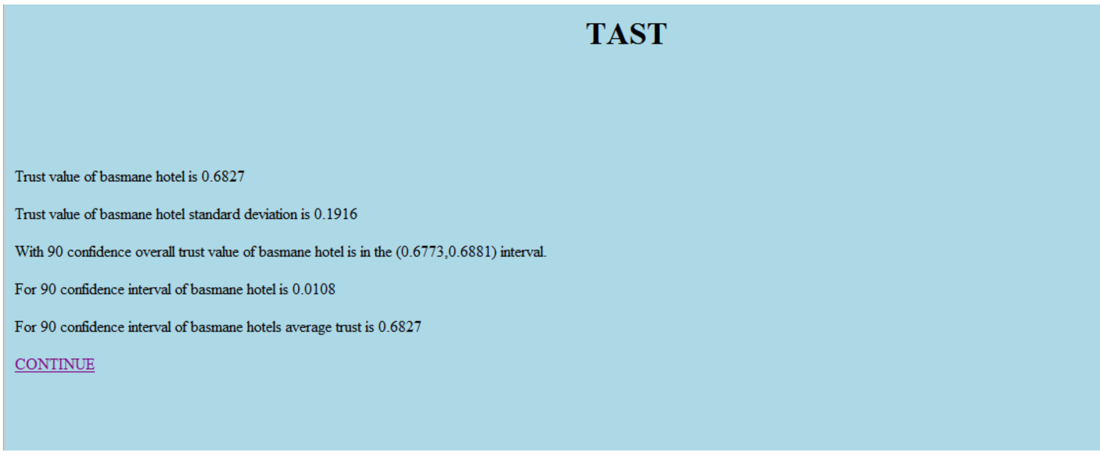
TAST

Popularity value of basmane hotel is 20.4801

Popularity value of basmane hotel standard deviation is 5.7492

Figure 6.25 Calculated popularity and standart deviation values

On this page calculated popularity and standart deviation values of the object for the selected time interval is displayed.When *continue* button clicked page shown in figure 6.26 is displayed.



TAST

Trust value of basmane hotel is 0.6827

Trust value of basmane hotel standard deviation is 0.1916

With 90 confidence overall trust value of basmane hotel is in the (0.6773,0.6881) interval.

For 90 confidence interval of basmane hotel is 0.0108

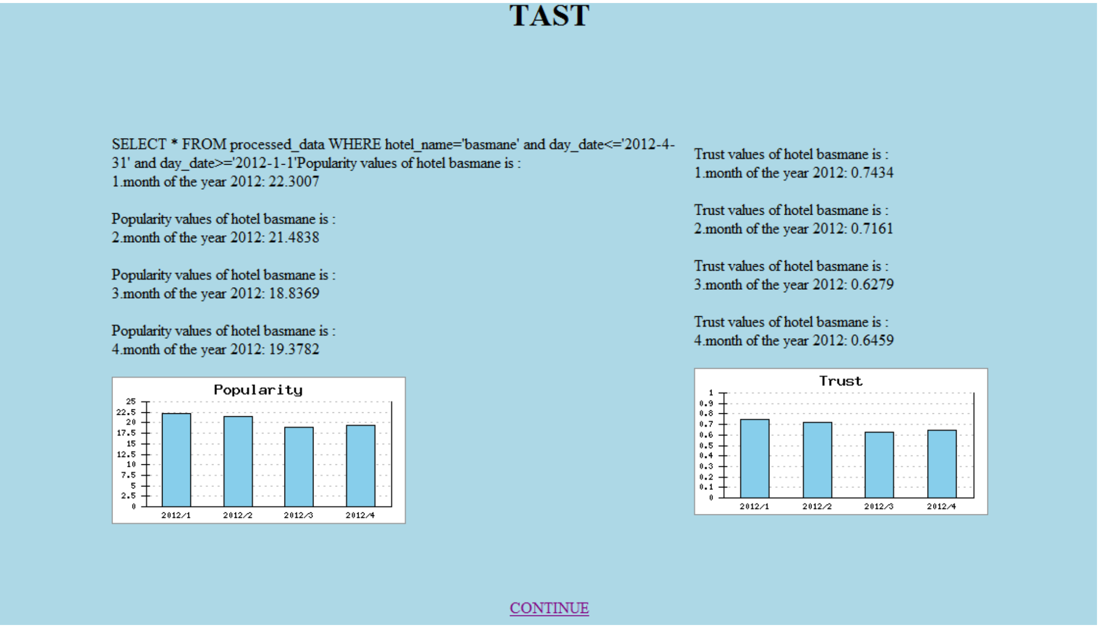
For 90 confidence interval of basmane hotels average trust is 0.6827

Figure 6.26 Calculated overall trust interval, standart deviation and average trust values

On this page calculated overall trust interval, standart deviationand average trust values of the object for the selected time interval is displayed.When *continue* button clicked to the homepage shown in figure 6.5 is displayed.

6.2.5 Graphical Representation of Popularity and Trust Variations

To obtain graphical representation of popularity and trust ariations of an object fifth option is selected on homepage shown in figure 6.5. Steps shown in figures from 6.21 to 6.23 are again applied and the page shown in figure 6.27 is displayed.


TAST

SELECT * FROM processed_data WHERE hotel_name='basmane' and day_date<='2012-4-31' and day_date>='2012-1-1'Popularity values of hotel basmane is :
1.month of the year 2012: 22.3007

Popularity values of hotel basmane is :
2.month of the year 2012: 21.4838

Popularity values of hotel basmane is :
3.month of the year 2012: 18.8369

Popularity values of hotel basmane is :
4.month of the year 2012: 19.3782

Trust values of hotel basmane is :
1.month of the year 2012: 0.7434

Trust values of hotel basmane is :
2.month of the year 2012: 0.7161

Trust values of hotel basmane is :
3.month of the year 2012: 0.6279

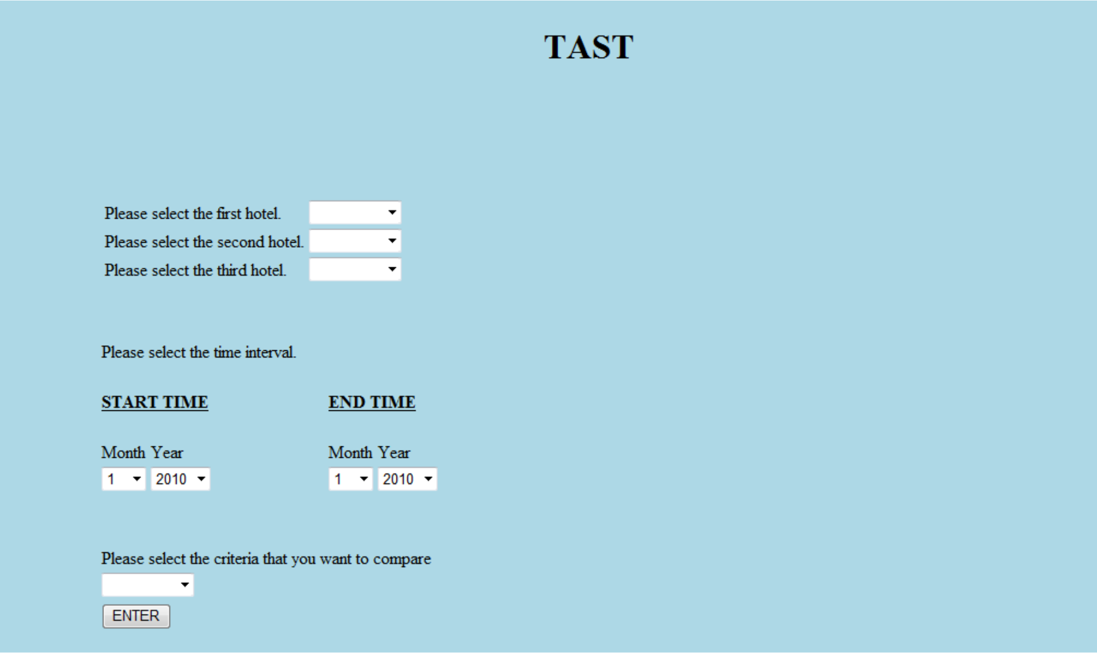
Trust values of hotel basmane is :
4.month of the year 2012: 0.6459

Figure 6.27 Calculated popularity and trust values for the selected months

On this page calculated popularity and trust values for the selected months are displayed numerically and graphically. By clicking continue to the home page shown in figure 6.5 is returned.

6.2.6 Comparison of Popularity and Trust Values of Objects

By selecting the last option and clicking the *enter* button the page shown in figure 6.28 is displayed. Three objects at the most can be selected in one time. The time interval in which comparison will be performed is selected by the drop down menu. Comparison criteria can be selected by using the drop down menu at the bottom of the page. Comparisons can be made as popularity or trust values. If popularity criteria is selected and enter clicked the page in figure 6.29 is diplayed. If trust criteria is selected and enter clicked the page in figure 6.30 is displayed. When continue buton clicked on these page, the home page shown in figure 5 is displayed.



TAST

Please select the first hotel.

Please select the second hotel.

Please select the third hotel.

Please select the time interval.

START TIME

Month Year

1 2010

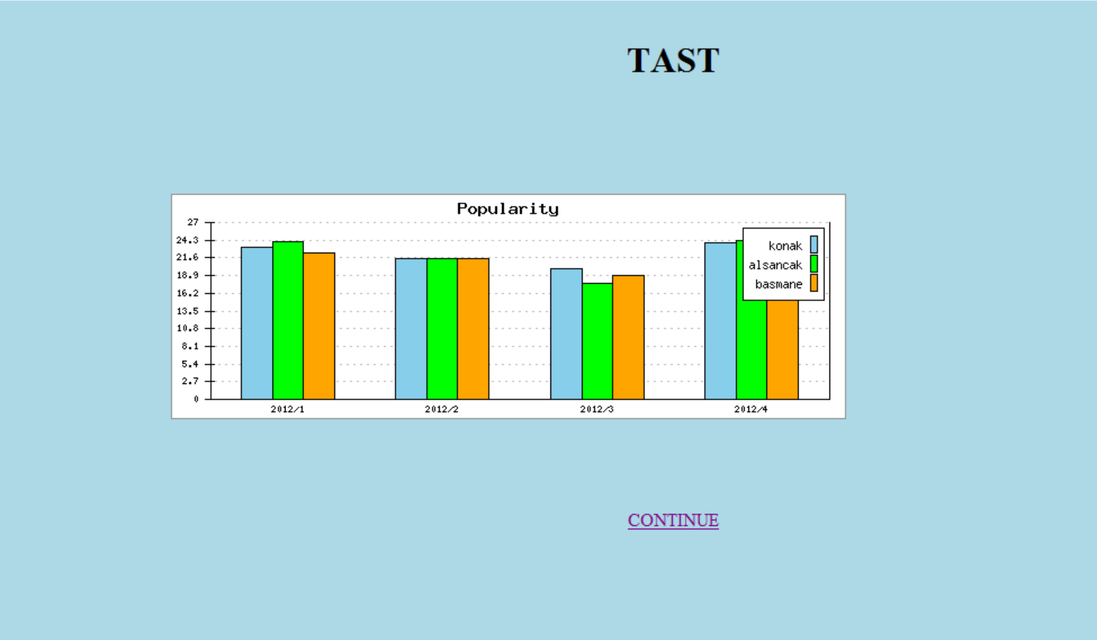
END TIME

Month Year

1 2010

Please select the criteria that you want to compare

Figure 6.28 Comparison of popularity and trust values of selected objects in selected time intervals



TAST

Figure 6.29 Graphical comparison of popularity values of objects

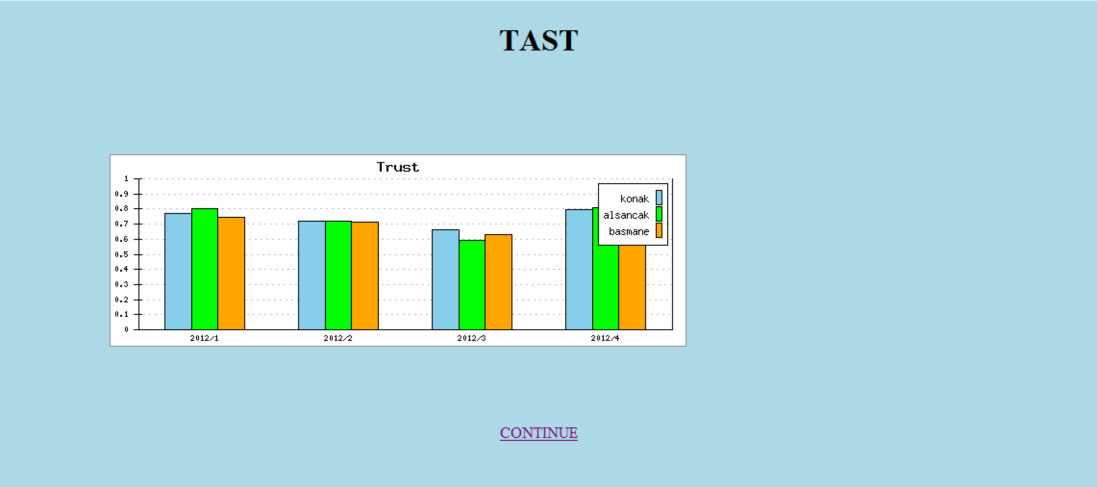


Figure 6.30   Graphical comparison of trust values of objects

6.2.7 Discussion About TAST

Some previous software developments like cTla trust evaluation of the trust value is done based on only reputation (Herrmann, 2006). It models reputation based trust as a decaying value, since recent information about an entity's reputation affects the level of trust to that entity more than past information.

Another research software named TRAVOS (Teacy et al, 2010) aims to calculate trust values by using the information from third party sources. They encountered the problem to reach the inaccurate information. Repeated interactions are required to distinguish reliable from unreliable sources.

TAST use data directly collected from customers by web based surveys. Customers are divided into classes by their common interests. Customer classes grade the features (or contexts of the organization) by their satisfaction. Collected data can be processed with time intervals which makes it possible to monitor tust variations. Survey can be applied to a group of organizations working on the same business field. TAST can compare competitor organization’s trust values in selected time intervals. TAST has the potentional to be developed as a commercial software.

CHAPTER SEVEN
CONCLUSIONS AND FUTURE WORK

This thesis examined formal trust assessment models based on bi-partite graphs. Main contributions of the thesis can be summarized as following:

- A formal model to assess the trust to the organizations in a specified context-set by using web-based survey data was developed. Addition of importance parameter to trust calculations and calculation of trust in real-number intervals by selected confidence probability were the main contributions.
- Trust and confidence propogation in trust chains were investigated. Propogation of confidence was here the main contribution.
- Trust and confidence propogation in service oriented systems were modeled. Propogation of confidence in service-oriented systems was again the main contribution in this model.
- A software tool called Trust Assesment Software Tool (TAST) was developed. This was a flexible program that can be applied to the organizations working in the same business-field. TAST calculated the trust assessments of the organizations in selected time intervals based on our hypothetical data. TAST made trust assessment comparisons by competitor organizations in selected time intervals.
- We also showed the applicability of our contributions by examples and case studies.

We plan to develop a new user interface for TAST. TAST has the potential of being used as a professional software for business and government organizations. We also plan to develop our model for service-oriented systems. This topic will be much more important in the future since demand for service-to-service interactions is rapidly increasing.

REFERENCES

Ajayi, O., Sinnott, R., & Stell, A. (2007). Trust Realisation in Multi-Domain Collaborative Environments. *IEEE Computer Society, Vol. 3, No. 12, 154-157.*

Andersen, R., Borgs, C., Chayes, J., Feige, U., Flaxman, A., Kalai, A., Mirokkoni, V., & Tennenholz, M. (2008). Trust Based Recommendation Systems: An Axiomatic Approach. *WWW2008, 199-208.*

Andert, D., Wakefield, R., & Weise, J. (2002). Trust Modeling for Security Architecture Development. *Tech. Rep., Sun Microsystems Inc.*

Artz, D., & Gil Y. (2007). A Survey of Trust in Computer Science and the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web Vol. 5, Issue 4, 227-239.*

Aziz, A., Singhal, V., & Balarin, F. (1995). It Usually Works: The Temporal Logic of Stochastic Systems. *Proceedings of the 7th International Conference on Computer Aided Verification, 155-165.*

Bahtiyar, Ş., Cihan, M., & Çağlayan, M.U. (2010). A Model for Security Information Flow on Entities for Trust Computation. *CIT 2010, 803-808.*

Bahtiyar, Ş., Cihan, M., & Çağlayan, M.U. (2009). An Architectural Approach for Assessing System Trust Based on Security Policy Specifications and Security Mechanisms. *SIN'09 Proceedings of the Second International Conference on Security of Information and Networks.*

Baier, A. (1986). Trust and Antitrust. *Ethics, 96, 231-260 (p. 242).*

Ball, R. N., Pultr, A., & Vojtechovsky, P. (2007). Colored Graphs Without Coloful Cycles. *Combinatorica, 407–427*.

Barber, K.S., Fullam, K., & Kim, J. (2002). Challenges for Trust, Fraud and Deception Research in Multi-agent Systems. *Trust Reputation and Security, 8-12.*

Bargh, M., Janssen, W., & Smit, A. (2002). Trust and Security in E-business Transactions. *Journal of e-Business Transactions.*

Ben-Gal, I., Ruggeri, F., Faltin, F. & Kenett, R. (2007). *Bayesian Networks.* Encyclopedia of Statistics in Quality & Reliability. Wiley & Sons.

Bertino, E., Ferrari, E., & Squicciarini, A.C. (2004). Trust-X: A Peer-to-Peer Framework for Trust Establishment. *IEEE Communications Surveys and Tutorials, Vol. 16, No. 7, 827-842.*

Blaze, M., Feigenbaum, J., & Lacy J. (1996). Decentralized Trust Management. *IEEE Symposium on Security and Privacy.*

Biskup, J., Hielscher, J., & Wortmann, S. (2008). A Trust and Property Based Access Control Model. *Electronic Notes in Theoretical Computer Science 197, 169-177.*

Budalakoti, S., DeAngelis, D., & Barber, K.S. (2009). Expertise Modeling and Recommendation in Online Question and Answer Forums. *International Conference on Computational Science and Engineering, 481-488.*

Burgess, M., Canright, G., & Monsen, K.E. (2004). A Graph Theoretical Model of Computer Security: From File Sharing to Social Engineering. *International Journal of Information Security, Vol. 3, No. 1, 70-85.*

Canfora, G., Costante, E., Pennino, C., & Visaggio, A. (2008). A Tree Layered Model to Implement Data Privacy Policies. *Computer Standarts and Intefaces, No. 30, 398-409.*

Carbone, M., Nielsen, M., & Sassone, V. (2003). A Formal Model for Trust in Dynamic Networks. *1st International Conference on Software Engineering and Formal Methods, 54-75.*

Carroll, J., Bizer, C., Hayes P., & Stickler P. (2005). Named Graphs, Provenance and Trust. *Proceedings of the 14th International Conference on World Wide Web WWW 05, 613-622.*

Castelfranchi, C., & Falcone, R. (2000). Trust is Much More than Subjective Probabilitiy: Mental Components and Sources of Trust. *Proceedings of the 33rd Hawaii International Conference on System Sources.*

Chen, T., Bu, T., Zhang, M., & Zhu, H. (2009). Max-Minimum Algorithm for Trust Transitivity in Trustworthy Networks. *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology Workshops , 62-64.*

Chen, G., Li, Z., Cheng, Z., Zhao, Z. & Yan H. (2005). A Fuzzy Trust Model for Multi-agent System. *In ICNC (3), 444-448.*

Christopher J.H., & Munidar P.S. (2010). Intertemporal Discount Factors as a Measure of Trustworthiness in Electronic Commerce. *IEEE Transactions* on *Knowledge and Data Engineering.*

Coleman, J.S. (1998). Social Capital in the Creation of Human Capital. *American Journal of Sociology (94), 95–120.*

Cremonesi, P., Garzotto, F., Negro, S., Papadapoulos A., & Turrin R. (2011). Comperative Evaluation of Recommender System Quality. *Proceedings of the Annual Conference on Human Factors in Computing Systems, 1927-1932.*

Dey, A.K. (2001). Understanding and Using Context. *Personal and Ubiquitous Computing Journal, Vol. 5, 4-7.*

Dutertre, B. (2001). Complete Proof Systems for First Order Interval Temporal Logic. *10th annual IEEE Symposiun on Logic in Computer Science*.

Dey, A.K. (2001). Understanding and Using Context. *Personal and Ubiquitous Computing Journal, Vol. 5, 4-7.*

Deutsch, M. (1958). Trust and Suspicion. *The Journal of Conflict Resolution, Vol.2, No. 4, 265-279.*

Diestel, R. (2000). *Graph theory* (electronic edition). Springer-Verlag New York.

Driscoll, J.W. (1978). Trust and Participation in Organizational Decision Making as Predictors of Satisfaction. *Academy of Management Journal, Vol. 21, 44-56.*

Esfendiari, B., & Chandrasekharan, S. (2001). On How Agents Make Friends: Mechanisms for Trust Acquisition. *Proceedings of the Fourth Workshop on Deception, Fraud and Trust in Agent Societies, 27-34.*

Gefen, D., Srinivasan, R., & Tractinsky, N. (2003). The Conceptualization of Trust, Risk and Their Relationship in Electronic Commerce: The Need for Clarifications. *Proceedings of the 36th Annual Hawaii International Conference on System Sciences.*

Gentle, J., Hardle, W., & Mori, Y. (2004). *Handbook of Computational Statistics.* Springer.

Grandison, T., & Sloman, M. (2000). A Survey of Trust in Internet Applications. *IEEE Communications Survey, Vol. 3, 2–16.*

Gross, J., & Yellen J. (2003). *Handbook of Graph theory*. CRC Press.

Guha, R., Kumar, R., Raghavan, P., & Tomkins, A. (2004). Propogation of Trust and Distrust. *Proceedings of the 13th International Conference on World Wide Web, 403-412.*

Hang, C., Wang, Y., & Singh, M.P. (2008). Operators for Propagating Trust and Their Evaluation in Social Networks. *AAMAS, 1485-1488.*

Hang, C. & Singh, M.P. (2009). Trustworthy Service Selection and Composition. *Proceedings of the 12th AAMAS Workshop on Trust in Agent Societies.*

Haque, M.M., & Ahamed, S.I. (2007). An Omnipresent Formal Trust Model (FTM) for Pervasive Computing Environment. *31st Annual International Computer Software and Applications Conference, COMPSAC, 49-56.*

Heitz, M., & König, S. (2009). Reputation in Multiagent Systems and the Incentives to Provide Feedback. *Information Systems Management Working Paper Series, Band 38, University of Bayreuth.*

Herrman, P. (2003). Trust-based Protection of Software Component Users and Designers. *Proceedings of the 1st International Conference on Trust Management, 75-90, LNCS 2692.*

Herrman, P. (2006). Temporal Logic-Based Specification and Verification of Trust Models. *iTrust, LNCS 3986, 105-119, Springer-Verlag Berlin Hiedelberg.*

Hines, W.W., Montgomery, D.C., Goldsman, D.M., & Borror, C.M. (2003). *Probability and Statistics in Engineering.* John Wiley & Sons Inc.

Holtmanns, S., & Yan, Z. (2006). Context-Aware Adaptive Trust. *Ambient Intelligence Developments Conference, 137-146.*

Hu, Y., Koren Y., & Volinsky C. (2008). Collaborative Filtering for Implicit Datasets. *Eighth IEEE International Conference on Data Mining, 263-272.*

Huang, Z., Chung, W., & Chen, H. (2004). A Graph Model for e-Commerce Recommender Systems. *JASIST, Vol. 55, No.3., 259-274.*

Jeffrey R. (2004). Subjective Probability: The Real Thing. *Cambridge University Press.*

Jøsang, A., Møllerud, P., & Cheung, E. (2001). Web Security: The Emperor's New Theol Armour. *ECIS.*

Josang, A. (2002). Subjective Evidental Reasoning. *IPMU.*

Jøsang, A., Gray, E., & Kinateder, M. (2003). Analysing Topologies of Transitive Trust. *Proceedings of the Workshop of Formal Aspects of Security and Trust (FAST*), *Pisa*.

Jøsang, A., Keser, C., & Dimitrakos, T. (2005). Can We Manage Trust? *Proceedings of the Third International Conference on Trust Management, 93-107.*

Jøsang, A., Pope, S., & Daniel, M. (2005). Conditional Deduction Under Uncertainty. *In Proceedings of the 8th European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty.*

Josang, A., Hayward, R., & Pope S. (2006). The Network Analysis with Subjective Logic. *Proceedings of the Australasian Computer Science Conference.*

Josang, A., Ismail, R., & Boyd, C. (2007). A Survey of Trust and Reputation Systems Online Service Provision. *Decision Support Systems, 43(2), 618-644.*

Knobloch, E., Leibniz, & Euler (1991). Problems and Solutions Concerning Infinitesimal Geometry and Calculus. *Conference on the History of Mathematics, 293-313.*

Koren, Y., Bel, R., & Volinsky C. (2009). Matrix Factorization Techniques for Recommender Systems. *IEEE Computer, Vol.42, No. 8, 30-37.*

Kuter, U., & Golbeck, J. (2007). *SUNNY:* A New Algorithm for Trust Inference in Social Networks Using Probabilistic Confidence Models. *Proceedings for the Twenty-Second AAAI Conference on Artificial Intelligence*.

Lamport, L. (2002). *The TLA+ Language and Tools for Hardware and Software Engineers.* Addison-Wesley.

Lang, B. (2010). A Computational Trust Model for Access Control in P2P. *Science China, 896–910.*

Li, J., Huai, J., & Hu, C. (2007). *PEACEVO:* A Secure Policy-Enabled Collaboration Framework for Virtual Organizations. *26th IEEE Symposium on Reliable Distributed Systems (SRDS), 199-208.*

Liu, C., Ozols, M.A. & Orgun, M.A. (2005). A Temporalized Belief Logic for Specifying the Dynamics of Trust for Multi-agent Systems. *LNCS, Vol. 3321, 142-156.*

Ma, J., & Orgun, M A. (2006). Trust Management and Trust Theory Revision. *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans, Vol.31, 451-460.*

Madigan, D., Mosurski, K., & Almond, G.R. (1997). Graphical Explanation in Belief Networks. *Journal of Computational and Graphical Statistics, Vol. 6, 160-181.*

Marsh, S., & Dibben, M.R. (2005). Trust, Untrust, Distrust and Mistrust – An Exploration of the Dark(er) Side. *3rd International Conference on Trust Management, iTrust*.

Massa, P. (2003). *Trust-aware Decentralized Recommender Systems.* PhD Research Proposal, Department of Information and Communication Technology, University of Trento.

Mayer R.C., Davis J.H., & Schoorman, F.D. (1995). An Integrative Model of Organizational Trust. *The Academy of Management Review, Vol.20, No.3, 709-734.*

McKnight, D.H., Cummings L., & Chervany N.L. (1998*).* Initial Trust Formation in New Organizational Relationships. *The Academy of Management Review , Vol. 23, No.3, 473-490.*

Mejia, M., Pena, N., Munoz, L., & Esparza, O. (2009). A Review of Trust Modeling in Ad-Hoc Networks. *Internet Research Vol. 19, No.1, 88-104.*

Michalakopoulos, M., & Fasli, M. (2005). On Deciding to Trust. *3rd International Conference on Trust Management, iTrust*.

Mistzal, B. (1996). *Trust in Modern Societies: The Search for the bases of Social Order.* Polity Press.

Moszkowski, B. (2007). Using Temporal Logic to Analyse Temporal Logic: A Hierarchical Approach Based on Intervals. *Journal of Logic and Computation Vol.17, No. 2, 333-409.*

Mui, L., Mohtashemi, M, & Halberstadt A. (2002). A Computational Model of Trust and Reputation for e-Business. *Proceedings of the 35th Annual Hawaii*

International Conference on System Science, vol.7 of HICSS'02, IEEE Computer Society.

Nam., J. (2009). A Trust Framework of Ubiquitous Healthcare with Advanced Petrinet Model. *1st International Conference on Electronic Healtcare, 122-129.*

Neisse, R., Wegdam, M. & Sinderen, M. (2006). Context-Aware Trust Domains. *EUROSSC 2006, LNCS 4272, 234 – 237.*

Neuman, W.L. (2000). *Social Research Methods: Qualitative and Quantitative Approaches (4th ed)*. Boston: Allyn and Bacon,

Olivieroa, F., Pelusoa, L., & Romano, S. (2008). Refacing: An Autonomic Approach to Network Security Based on Multidimensional Trustworthiness. *Computer Networks, Vol. 52, 2745–2763.*

Orgun, M.A., & Liu C. (2006). Reasoning about Dynamics of Trust and Agent Beliefs. *IEEE International Conference on Information Reuse and Integration, 105-110.*

Purser, S. (2001). A Simple Graphical Tool For Modelling Trust. *Computers & Security No. 20, 479-484.*

Rasmusson, L., & Janson S. (1996). Simulated Social Control for Secure Internet Commerce. *Proceedings of the 1996 Workshop on New Security Paradigms.*

Ray, I., & Chakraborty, S. (2009). An Interoperable Context Sensitive Model for Trust. *Journal of Intel on Information Systems No.32, 75-104,*

Raya, M., Papadimitratos, P., Gligor, V., & Hubaux, J. (2008). On Data-Centric Trust Establishment in Ephemeral Ad-Hoc Networks. *IEEE INFOCOMM, The 27th Conference on Computer Communications.*

Schafer, J.B., Konstan J., & Riedl, J. (2004). Recommender Systems in e-Commerce. *ACM Conference on e-Commerce, 158-166.*

Sebater i Mir, J. (2003). *Trust and Reputation for Agent Societies.* Doctoral Graduation Thesis, Artificial Intelligence Resarch Institute, University of Barcelona.

Schafer, J.B., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative Filtering Recommender Systems. *The Adaptive Web: LNCS4321, 291-324.*

Smith, C. (2001). Trust and Confidence: Possibilities for Social Work in High Modernity. *British Journal of Social Work 31, 287–305.*

Sun, Y., Han, Z., & Liu, L. (2008). Defense of Trust Management Vulnerabilities in Distributed Networks. *IEEE Communications Magazine, 112-119, February.*

Teacy, L., Patel, J., Jennings, N.R., & Luck, M. (2006). TRAVOS: Trust and Reputation in the Context of Inaccurate Information Sources. *Autonomus Agents and Multi-Agent Systems, Volume 12, 183-198.*

Theodorakopoulos, G., & Baras, J.S. (2006). A Testbad for Comparing Trust Algorithms. *25th Army Science Conference.*

Thiagarajan, K., Raghunathan, A., Natarajan, P., Poonkuzhali, G., & Ranjan R. (2009). Weighted Graph Approach for Trust Reputation Management. *World Academy of Science, Engineering and Technology, No.56, 830-836.*

Trcek, D. (2009). A Formal Apparatus for Modeling Trust in Computing Environments. *Mathematical and Computer Modeling, Vol.49, 226-233.*

Yan, Z. (2007). *Trust Management for Computing Platforms.* Ph. D. Thesis, Department of Electrical and Communication Engineering, Hesinki University of Technology.

Yang, Y., Brown, L., Lewis, E., & Newmarch, J. (2002). W3 Trust Model: Evaluating Trust and Transitivity of Trust of Online Services. *Proceedings of the International Conference on Internet Computing, 354-362.*

Yao, D., Shin, M., Tamassia, R., & Winsborough W. (2005). Visiulazation of Trust Negotiation. *In Proceedings of the 1st Workshop on Visiulazition for Computer Security, Vizsec*.

Yolum, P. (2003). Properties *of Referral Networks: Emergence of Authority and Trust*. Doctoral Graduation Thesis, Department of Computer Science North Carolina State University.

Yolum, P., & Singh, M.P. (2003). An Agent-Based Approach for Trustworthy Service Location. *In Proceedings of the 1st Workshop on Agents and Peer-to-Peer Computing (AP2PC), 45–56, LNAI 2530, Springer-Verlag*.

Yolum, P., & Singh, M.P. (2004). Service Graphs for Building Trust. *In Proceedings of the 12th International Conference on Cooperative Information Systems, 509-525.*

Wang, Y., & Singh, M.P. (2007). Trust Representation and Aggregation in a Distributed Agent System. *IJCAL, 1551-1556.*

Wang, Y., & Singh, M.P. (2010). Evidence-Based Trust: A Mathematical Model Geared for Multiagent Systems. *ACM Transactions on Autonomous and Adaptive Systems, Vol. 5, No. 3, 1-25*.

Weeks, S. (2001). Understanding Trust Management Systems. *IEEE Symposium on Security and Privacy, 94-105.*

Wei-Peng, L., & Ju, H. (2008). A Formal Model of Trust and Security for Task-Oriented Information System. *Proceedings of The International Symposium on Electronic Commerce and Security, 502-506.*

Xiu, D., & Liu, Z. (2005). A Formal Definition for Trust in Distributed Systems. *Information Security: 8th International Conference, 482-489.*

Zejda, D. (2010). From Subjective Trust to Objective Trustworthiness in On-line Networks: Overview and Challenges. *Journal of Systems Integration, Vol. 1, No. 1-2, 3-15.*

Zia, A.Z. (2008). Reputation-based Trust Management in Wireless Sensor Networks. *In the Proceedings of the Fourth International Conference on Intelligent Sensors, Sensor Networks and Information Processing ISSNIP, 163-166.*

Zhou, R., & Hwang, K. (2007). Powertrust: A Robust and Scalable Reputation System for Trusted Peer to Peer Computing. *IEEE Transactions on Software Engineering, Vol. 18, No. 4, 460-473.*

APPENDİCES

A: Sample Data Used for Hotel Trust Assesment Database

User ID	User Type	Hotel ID	Value	Imp.	Room	Imp.	Location	Imp.	Clean	Imp.	Service	Imp	Sleep	Imp	Date	Time
100	Business	Basmane	5	3	5	3	5	3	5	3	5	3	4	3	02:01:2012	01:01:00
101	Business	Alsancak	4	3	5	3	4	3	5	3	5	3	5	3	02:01:2012	02:01:00
102	Solo	Basmane	4	3	5	3	5	3	5	3	5	3	5	3	02:01:2012	03:01:00
103	Business	Konak	4	2	4	3	5	3	5	3	5	3	5	3	02:01:2012	04:01:00
104	Family	Basmane	5	3	4	2	5	3	5	3	5	3	5	3	02:01:2012	05:01:00
105	Business	Alsancak	5	3	4	2	5	3	5	3	4	3	4	3	02:01:2012	06:01:00
106	Couples	Alsancak	5	3	4	3	5	3	5	2	4	2	4	2	02:01:2012	07:01:00
107	Business	Basmane	5	3	5	3	5	3	4	3	4	2	4	3	02:01:2012	08:01:00
108	Friends	Konak	5	3	5	3	5	3	4	3	4	3	4	3	02:01:2012	09:01:00
109	Business	Konak	5	3	5	3	5	3	3	3	4	2	5	3	03:01:2012	01:01:00
110	Friends	Basmane	5	3	5	2	5	2	5	3	4	3	5	3	03:01:2012	02:01:00
111	Couples	Alsancak	4	3	5	2	5	2	5	3	5	2	5	3	03:01:2012	03:01:00
112	Business	Konak	4	3	5	2	5	2	5	3	5	1	5	3	03:01:2012	04:01:00
113	Solo	Alsancak	4	2	5	2	4	2	5	3	5	2	5	3	03:01:2012	05:01:00
114	Business	Basmane	4	2	5	3	4	3	5	3	3	3	5	2	03:01:2012	06:01:00
115	Business	Konak e	4	2	5	3	4	3	5	1	5	3	4	3	03:01:2012	07:01:00
116	Couples	Basmane	4	2	5	3	4	3	5	2	5	3	4	3	03:01:2012	08:01:00
117	Business	Konak e	4	2	5	3	4	3	4	3	5	3	4	3	03:01:2012	09:01:00
118	Business	Basmane	4	2	4	3	4	3	4	3	5	3	4	3	04:01:2012	01:01:00
119	Business	Alsancak	4	2	4	3	4	3	4	3	5	3	5	3	04:01:2012	02:01:00
120	Solo	Basmane	3	3	4	3	3	3	5	3	5	2	5	2	04:01:2012	03:01:00
121	Business	Konak	5	3	4	2	3	3	5	3	3	2	5	2	04:01:2012	04:01:00
122	Family	Basmane	5	3	4	2	3	3	5	3	4	2	4	1	04:01:2012	05:01:00
123	Business	Alsancak	5	3	4	2	3	2	3	3	5	1	3	2	04:01:2012	06:01:00
124	Friends	Alsancak	5	3	4	3	3	2	4	3	5	1	4	2	04:01:2012	07:01:00
125	Business	Basmane	5	3	4	3	4	2	4	3	5	2	5	3	04:01:2012	08:01:00
126	Solo	Konak	5	3	4	3	4	1	4	1	5	3	5	3	04:01:2012	09:01:00
127	Business	Basmane	4	3	4	3	5	3	5	2	5	3	5	3	05:01:2012	01:01:00
128	Family	Konak	4	3	4	1	5	3	5	3	5	3	5	3	05:01:2012	02:01:00
129	Business	Basmane	4	3	4	3	5	3	5	3	5	3	5	3	05:01:2012	03:01:00
130	Solo	Basmane	4	1	4	3	5	3	5	3	5	3	5	3	05:01:2012	04:01:00
131	Business	Konak	4	3	3	3	5	3	5	3	5	3	5	3	05:01:2012	05:01:00
132	Friends	Alsancak	4	3	5	3	5	3	5	3	5	2	5	3	05:01:2012	06:01:00
133	Family	Basmane	4	3	5	3	5	3	5	3	5	3	5	3	05:01:2012	07:01:00
134	Business	Konak	3	3	5	3	5	3	5	3	5	2	4	3	05:01:2012	08:01:00
135	Couples	Basmane	5	3	5	3	5	3	5	3	4	3	4	3	05:01:2012	09:01:00
136	Business	Konak	5	3	5	3	5	3	4	3	4	2	4	3	06:01:2012	01:01:00
137	Family	Alsancak	5	3	5	3	5	3	4	3	4	3	4	3	06:01:2012	02:01:00

B: Z_{α} Values

Confidence	z-alpha
99	2.58
98	2.33
97	2.17
96	2.05
95	1.96
94	1.88
93	1.81
92	1.75
91	1.70
90	1.65
89	1.60
88	1.55
87	1.51
86	1.48
85	1.44
84	1.41
83	1.37
82	1.34
81	1.31
80	1.28