COMPUTATIONS IN SOCIAL NETWORK

A thesis submitted
to Kent State University in partial
fulfillment of the requirements for the
degree of Masters of Science

by
Sajid S Shaikh
July 2007

Thesis written by

Sajid S Shaikh

B.E, Pune University, 2000

M.S, Kent State University, 2007

Approved by

Dr. Javed I Khan _____, Advisor

Dr. Robert A Walker _____, Chair, Department of Computer Science

Dr. Jerry Feezel _____, Dean, College of Arts and Sciences

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

A society is a grouping of individuals, which is characterized by common interest and may have distinctive culture and institutions. As the members of a society grow, there is a trend of smaller communities being formed within the society. *Thus, a community is a tighter and more cohesive social entity within a larger society, due to the presence of a unity of will*. As the number of such communities increases people start to limit their social interactions to within their community forming a social network. *Hence, a social network is a social structure made up of nodes that are tied by one or more specific types of relations and relationships are social associations, connections, or affiliation between two or more people*. As the volume of this inter-social network interaction increases each member unknowingly gathers extensive information about his/her peers. Thus, over a period these social networks inadvertently become a reservoir of social knowledge about its members. This knowledge base becomes very essential when an outsider wants to deal with a member this social network. He/She can investigate the knowledge base and draw an approximate social profile of a member even without meeting him/her. Depending upon the social profile one can decide whether to interact with a certain individual or not, hence considerably reducing risky interactions.

Similarly, Internet today is like a virtual society serving an array of different people in variety of different ways. In the recent past like-minded people have started coming together to form virtual communities on the World Wide Web. The members of these communities form a social network through their interactions with each other on the world wide web. As the web increasingly becomes an influential part of people's lives, the distinction between the actual and the virtual social network is rapidly fading. One can know a great deal about a person without physically meeting or talking to him. Thus the internet has made social interactions between individuals separated by vast geographical distances possible.

In this theses we are to point out how boundary between the real and the virtual world is being faded further with the advent of the various social networking platforms. The individuals under consideration are from the real world but the data regarding their social interaction and relationships is gathered from their profiles on social networking platforms. The applications which have emerged through the computations on this data can be used in to solve the daily society specific problems an individual faces. A few examples of these problems would be finding social status of individuals, finding like minded people, determining the centers of influence in a community, determining the trustworthiness of individuals etc. Thus a social network based computation is the usage of an individuals social data to provide a guideline to optimize his/her interactions in both the societies (real/virtual). Give examples

An individual is progressively depending upon social networking platforms for companionship, advice, entertainment, education etc. Social network platforms such as Orkut® [4], Yahoo360®, MySpace®, LinkedIn® are providing social network interaction services. Orkut is a website designed specifically for friends and family. The whole thrust of Orkut is to make the conversation with friends and family more upbeat and fun. It further allows members to create

communities, so that like-minded people can meet up and have lively and engaging discussions. Orkut's use as a social tool is complex, because various people frequently try to add strangers to their pool of friends, more often than not just to increase the number indicating their number of friends next to their name in their profile. LinkedIn is example of another social networking service geared towards professionals. Even though the primary objective of many of these websites is to connect people over the Internet, the audience they serve is often dissimilar to each other and they offer variants of privacy settings and communication tools.Table-1 lists some of the leading community networks of today. Figure 1 provides a snapshot portal from two sites. These portals represent just one node in a vast network comprised of millions of nodes. The graphic identifies the link types provided by these services. Many of these services themselves provide a host of social network powered communication tools to the community. Services such as Google's Gmail® have structured their expansion of social network. In addition, various peer-to-peer networks are weaved in the fabric of social network. At the very heart of these systems is an extensive relationship network. Very powerful applications are conceivable from the global relationship information available in them. Using an individual's social network profile on Orkut and his scrapbook one can make a calculated guess about the relationship network of the individual. The scrapbook can also help one to determine the strength of these relationships. If one wants to find out the set of close friends of his boss, one can easily achieve this through the information available on Orkut. Similarly, if one wants to know about his boss's interests the community section on the boss's profile can give him some sort of starting point. Thus even though this information seems to be very naïve, a concerted social network computing can help to derive relationships which are not obvious on surface but do exist. Similarly, LinkedIn can be used to find out the firms and organizations a person has been affiliated in the past. This information can be used to find a set of people who are likely to have an influence on the individual.

We can develop strategies to query the social knowledge base for establishing various social properties and design a system that can assist peers in answering questions about an individual's social standing. Almost all the online communities- ranging from buyers, sellers, or auctioneers of e-commerce-sites, millions of peer-to-peer file sharers, to the brigade of editors in wiki-sites- all need a Relationship Algebra to define and measure relationships. In the real world, we notice that there are various social factors, which when considered together help in profiling an individual and in predicting his future behavior. The knowledge of one helps in deriving the other or in strengthening its value. Such two related social factors are trust and reputation. Trust has been defined in various ways in the literature. However, the following two definitions encompass the numerous flavors of trust. Gambetta's [1] trust definition is that "Trust is a skewed probability on the basis of which an individual expects another individual to behave in a certain way". McKnigth & Chervany [2] define trust in a somewhat different way. Their definition of trust is with respect to decision-making. According to them "Trust is the degree of security an individual feels with respect to another individual even though he knows that there is risk involved". Normally a person tends to trust another person if that person has a good reputation in the community. One instinctively avoids dealing with people having bad reputations. This kind of behavior is intrinsic to the way humans interact within and outside their community. An individual's standing in society is dependant upon and defined by his reputation. According to the Merrian-Webster dictionary, "Reputation is the overall quality or character as seen or judged by people in general". This definition reflects the influence of an individual's social network on his reputation. Broadly, speaking reputation is essentially the community's collective view about an individual. The people an individual interacts with in his day-to-day existence essentially represent his community, which forms his social network. Thus, reputation is a cooperative measure of trustworthiness based on the opinions expressed by members of an individual's social network. Another interest-

ing social law is Influence. Miriam-Webster dictionary defines influence as “the act or power of producing an effect without apparent exertion of force or direct exercise of command” or “to affect or alter by indirect or intangible means”. In chapter 12 of Canadian Organization Behavior [3], the authors present the various types of influence. The types of influence are Silent Authority, Assertiveness, Exchange, Coalition Formation, Upward Appeal, Persuasion and Information Control. Through our survey we have found that, a number of factors affect influence. Here we present the ones, which were the most dominant. We call them the I-factors. The I-factors are of different types, a few are individual’s properties such as age, location while others depend upon the interactions between individuals, such as frequency of contact and trust. A few I-factors depend upon the relationship between the individuals, such as type of relationship while others depend upon the behavior of individuals in community such as reputation and trustworthiness.

Relationship algebra seems to be the need of the hour for any kind of community like activity where people have to interact with strangers. The need becomes greater in case of online activities such as feedback forums, expert sites, product review sites, discussion forums , ecommerce sites and most importantly in social networking platforms. At present, the reputation systems have been deployed on two main architectures: centralized and distributed. One of the logical environments for the deployment of relationship algebra is peer-to-peer (P2P) networks. The fastest growing application of P2P networks are the Social Networking Platforms such as Orkut, LinkedIn, MySpace etc.

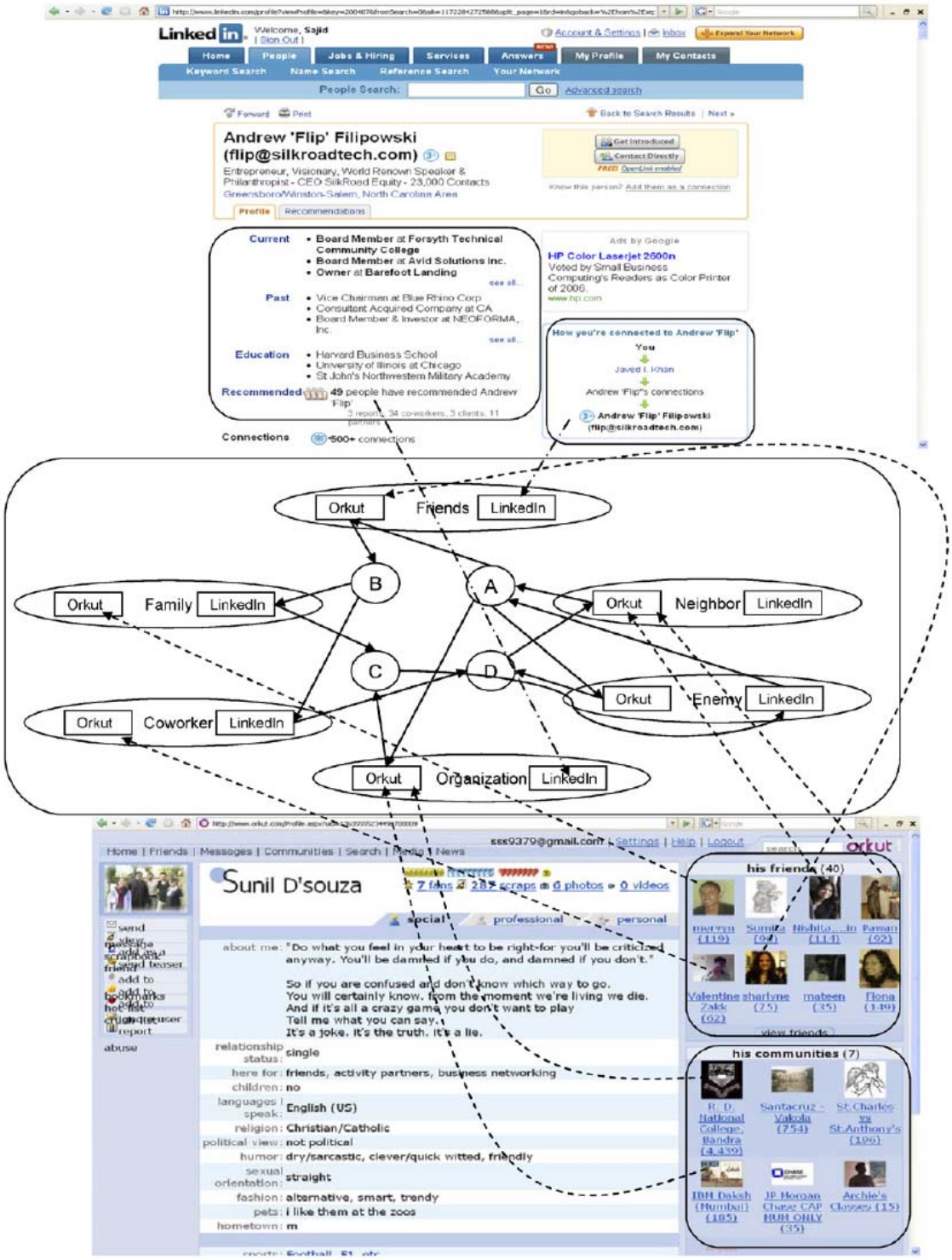


Figure 1 : The top part shows a LinkedIn web page, the middle part shows a social networks schema graph and the bottom part show an Orkut page.

Network	Members	User Base
Orkut	22,000,000	Designed specifically for friends and family
LinkedIn	6,000,000	Designed for professionals and adults.
MySpace	54,000,000	Used primarily for entertainment and blogging
Sporzoo	2,000,000	Real estate investors and professional
SelectedMinds	1,000,000	Corporate social networking

Table 1 : Survey of Some Current Social Networks

Thus, we can sense that social networking are one of the latest and fastest growing phenomena of the Internet. The websites providing social networking services are fast becoming an important cog in the borderless world of Internet. One can view them as digital town squares where different kind of people having varied interests can interact with each other. In the real world, individual's social interaction is based upon certain cognitive algorithms, which uses the social factors as an input and provides output that helps the individual in taking decisions with respect to his social behavior. Thus by using the social factors and various algorithms, which at first might seem fuzzy, individuals makes thousands of decisions on a daily basis. We call this amalgamation of factors and algorithms as social interaction mechanics. Thus, we see that one's decision to interact with someone is based purely on these mechanics. This mechanism is referred repeatedly even while making trivial decisions. Even though one might think that these algorithms are very fuzzy , in this thesis we have actually given a canonical classification of the algorithms and through our sample application shown how they correspond to the classical algorithms

If we have to duplicate these abstract values into a computable form, we need some kind of formulation, which clearly defines a framework to determine these factors. A completely new

range of powerful social network based applications –which can be called society applications are conceivable based on the social network assets growing under these communities. In this thesis, we expose how this vast knowledge base can be used in some real-life like applications.

In our work, we have outlined a framework to represent and reason with the general case of social relationship network. This basic framework called as the relationship algebra is used to define the relationship between various social nodes. The algebra consists of mainly two systems, the relationship reasoning system (RRS) and the relationship quantification system (RQS). The RRS is used to define the various relationships that exist in a social network, whereas the RQS is used to derive the strength of these relationships. This framework can be used to carry out various basic forms of analysis on the social network's knowledge base. In this thesis, we have proposed a canonical classification of these analyses. We have classified them as Social Profile Mining, Social Fabric Analysis, Social Linkage Analysis, Social Ranking Analysis and Placement within a Community. Further research into this field may lead to discovery of other classes of analysis. These classifications are not the definitive list of computations possible on a social network. We have supported our claim by providing algorithmic structure and computational solutions for each class of computation.

CHAPTER 2

RELATIONSHIP ALGEBRA

In this chapter, we introduce the Relationship algebra. The algebra consists of the relationship reasoning system (RRS) and the relationship quantification system (RQS). RQS can be used to derive social factors of an entity in a social network setting. Social factors can be defined as the corner stones based on which a society defines the social standing of an individual. A few social factors are reputation, trust, influence, status etc. We have demonstrated the social factor computation using the relationship algebra by deriving reputation, one of the fundamental social factors. In the process, we have also presented the various factors that affect social factors and have provided a canonical classification of the various classes of RQS. To demonstrate the robustness of our system we provide the behavior of RQS when exposed to different kinds of attacks.

A number of reputation computation systems have been proposed [5, 6, 7, 8, 9, 10, 11]. However, each of these solutions targets a very specific area and has been designed to serve only their domain. None of these solutions is customizable, so that they can be tweaked to be used in varied domain. Through our research, we present a computation system, based on social factor schema, which can be used to compute reputation in varied environments.

2.1. Representation

A society is comprised of unique social entities. Each unique entity (E) is represented by an entity ID. Entities in this world are however, organized as members of various sets. Lets us consider a publication network. There are sets such as author (A), paper (P), journals (J), reviewer (R), etc. An entity can be member of multiple sets. For example, individual 'Andrew' can be a member of an author set as well as of reviewer set. Members have also membership index in each set. The membership index of an entity does not have to be the same between sets. In a way, all objects in this world are members of the super set E. In this world–, various pairs of sets can have relationship. For example, papers have authors, i.e., set A and set P have relationship author-to-paper. Thus, a member in set A may or may not be an author-to-paper relation with each member of the paper set P.

Let A is a set (vector) of members of set author, and P is a set (vector) of members of type papers, then the cross product $M^r = AxP$ is the matrix of author-to-paper relationship, we call it relation matrix. Each element m_{ij} represents the strength of relationship. In real valued strength if m_{ij}= R it represents individual a_i has an author-to-paper relationship to paper p_j, or m_{ij}=0 indicates the absence of this relationship between the two.

We will use the notation M_i, - to present the i-the row of matrix M. It is a relationship statement about i-th member of A and says who in P are related with i-th member. We will use the notation M_I-j to present the j-the column of matrix M. It is a relationship statement about j-th member of set P and says who in A are related to this j-th member. Two sets can have more than one kind of relationship, each represented by a separate relation matrix M. We use the superscript r to denote the specific relationship of connecting sets.

2.2. Reputation Reasoning System

Now, we define a set of operations on relation matrices. If A is a relationship matrix then we define following semantic operators:

(i)Equivalence : Two relations are set to be equivalent if they are semantically same. If a matrix M denotes the relationship between i and j and if a semantically same relationship exists between k and j denoted by matrix N then the two matrices M and N are equivalent. For example the relation an individual's biological brother's biological father is semantically same to the biological relationship.

(ii) Synthesis : It is a method of inferring the relationship between two nodes which are not directly connected by using the directly connected relationships. For example if we consider three nodes i, j and k and if i is j's son and k is j's brother, then using synthesis we can derive that a nephew relationship exits between i and k.

(iii) reflection : Reflection means if i is related to j then j is also related to i. The matrix transpose is obtained by interchanging rows and columns. Thus if the elements of a matrix M denote the relationship from i to j then its' transpose M^T denotes the relationship from j to i . For example if i is j's husband then j is I's spouse. Thus spouse is the reflection relationship for husband.

(iv) exclusion : The exclusion operation is used for removal of relationship between two nodes. The exclusion operation is an element-wise subtraction of two matrices. For example if i is the grandson of j then there exist a relationship between i and j. When j dies we need to remove the relationship since j no longer exists in the network.

(v) semantic inverse : Semantically no functional form is known to compute the relation matrices for semantic inverse. For example if i trust j by an amount x then we can't say for sure

that i distrusts j by an amount y-x. The presence of certain relationship does not always mean the existence of it's semantic inverse.

We also define the following set operations:

(i) Intersection : The intersection of two sets A and B is the set that contains all elements of A that also belong to B but no other elements.

(ii) Dediagonalization: The dediagonalization operation sets all the elements in the diagonal of the relationship matrix to zero. This operation is applied when we do not want to consider self-to-self relations.

(iii) Set union: The union of a collection of sets A and B is the set that contains everything that belongs to any of the sets, but nothing else.

(iv) quantization. The quantization operation is used when we want to categorize relationship as existent or nonexistent depending upon its strength. For example if matrix M represents i relationships with each element value representing the strength of the relationship , then by using quantization we can say that relationships above a certain particular value would be considered while the ones below it wont be considered. Thus the resultant matrix after quantization has elements which either have a value of 1 or zero.

We introduce the following notation to denote the above operations. Equivalence is denoted by $A=B$, synthesis by AXB, reflection by A^T, absence by $\overline{A}$, semantic inverse by $\tilde{A}$. Following notations are used for element to element operations on relationship matrices : intersection of two relations by $A \otimes B$, union of two relations by $A \oplus B$, exclusion of related set B from related set A by $A \Theta B$, dediagonalization $\hat{A}$, and quantization $\lceil A \rceil$. The

above operations set now enables us to define, track, infer, and analyze various complex social relationships, and their interplays.

Operation	Symbol	Explanation
Column Extraction	**Ψ**	$\Psi_i^X(M)$ *where* X *is* $M_{ij} \geq \mu$ or X *is* $M_{ij} < \mu$
Row Extraction	**ρ**	$\rho_j^X(M)$ *where* X *is* $M_{ij} \geq \mu$ or X *is* $M_{ij} < \mu$
Max Row	**ξ**	$\xi_j^X(M)$ *where* X *is* $M_{ij} > \forall M_{kj}$
Max Column	**Φ**	$\phi_i^X(M)$ *where* X *is* $M_{ij} > \forall M_{ik}$
Zero Column	**θ**	$\theta^X(M)$ *where* X *is* $M_{ij} = 0$ *and* $0<i<n$

Table 2 : Set Operations

Operation	Symbol	Explanation
Equivalence	$A=B$	$a_{ij}=b_{ij}$
Row Extraction	$R=A^T$	$r_{ij}=a_{ji}$
Synthesis	$S=AxB$	$s_{ij}=\sum_{r=1}^{n} a_{ir}\ b_{rj}$
Intersection	$E=A\otimes B$	$e_{ij}=a_{ij}\cap b_{ij}$ *where* $0<i<n$, $0<j<m$
Union	$U=A\oplus B$	$u_{ij}=a_{ij}\cup b_{ij}$ *where* $0<i<n$, $0<j<m$

Exclusion	$X = A \Theta B$	$x_{ij} = a_{ij} - b_{ij}$ $where\ 0<i<n\ ,\ 0<j<m$
Dediagonalization	$\hat{A}$	$A_{ij} = \lfloor 0\ if\ i == j \rfloor$ $where\ 0<i<n\ ,\ 0<j<m$
Quantization	$\lceil A \rceil^{\mu}$	$a_{ij} = \begin{bmatrix} 1\ if\ a_{ij \geq \mu} \\ 0\ if\ a_{ij < \mu} \end{bmatrix}$ $where\ 0<i<n\ ,\ 0<j<m$

Table 3 : Relationship Operations

2.2.1 Set Algebra

Column Extraction (Ψ)

Given a matrix M_{ij} such that M_i are its rows ($0< i < n$) and M_j are the columns ($0 < j < m$). The set operation Ψ determines the extraction set for each M_i.

$$\Psi_i^X(M)$$
$$where\ X\ is\ M_{ij} \geq \mu \text{ or } X\ is\ M_{ij} < \mu$$

Row Extraction (ρ)

Given a matrix M_{ij} such that M_i are its rows ($0< i < n$) and M_j are the columns ($0 < j < m$). The set operation ρ determines the extraction set for each M_j.

$$\rho_j^X(M)$$
$$where\ X\ is\ M_{ij} \geq \mu \text{ or } X\ is\ M_{ij} < \mu$$

Max Row (ξ)

Given a matrix M_{ij} such that M_i are its rows ($0< i < n$) and M_j are the columns ($0 < j < m$). The set operation ξ determines the row M_i for M_j such that M_{ij} is highest value among all M_{kj} where $0 < k < n$

$$\xi_j^X(M)$$
$$where\ X\ is\ M_{ij} > \forall M_{kj}$$

Max Column (Φ)

Given a matrix M_{ij} such that M_i are its rows ($0 < i < n$) and M_j are the columns ($0 < j < m$). The set operation Φ determines the column M_j for row M_i such that M_{ij} is highest value among all M_{ik} where $0 < k < m$

$$\phi_i^X(M)$$
$$where\ X\ is\ M_{ij} > \forall M_{ik}$$

Zero Column(θ)

Given a matrix M_{ij} such that M_i are its rows ($0 < i < n$) and M_j are the columns ($0 < j < m$). The set operation θ determines each column M_j such that all the elements in M_j are equal to 0

$$\theta^X(M)$$
$$where\ X\ is\ M_{ij} = 0\ and\ 0 < i < n$$

2.3. Reputation Quantification System

Now we present the quantification system of the relationship algebra. Our goal is develop a social factor estimation function, which is generic and at the same type customizable such that it can mimic various models of local social factor estimation, which are encountered in real life. This is followed by a discussion of how this framework can be used to derive reputation. The discussion highlights the various factors that influence the reputation of a peer and towards the end; we present a mathematical formulation for quantifying reputation.

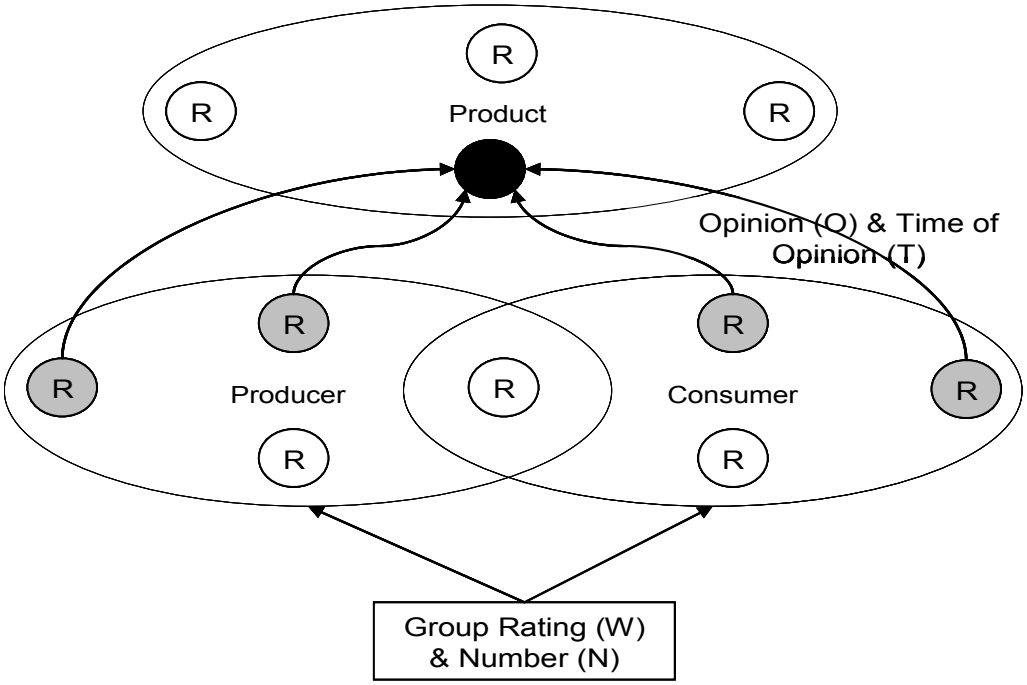


Figure 2 : Set Based Of Any Environment

Social factors are estimated in social setup. However, various social transactions are the basis for this evaluation process. Any transaction involves three parties: producer, product, and consumer. Each of the transactions occurs in a communal context. A particular product is sold repeatedly- but perhaps to different consumers, perhaps by different producers. Similarly, a consumer buys various products. Thus, there is a set of consumers, a set of producers and set of products. Hence, these transactions collectively build up a memory about a target individual and this is estimated using RQS. Now we present how we have applied RQS for estimating reputation. Any transaction creates a six way update of reputation estimation.

Figure 2 illustrates one such transaction. The producer and the consumer sets are expressing their opinions about a product in the product set. A generic reputation function seems to be based on various individual and group properties. However, depending upon the environment of deployment some of the individual and group properties would be included while others omitted when quantifying the reputation of an individual.

Generally, the reputation of a peer indicates the level of trust his community has in him. The reputation is dependant upon the kind of social interactions an individual has with his fellow peers. The interacting peers express their satisfaction or dissatisfaction by providing an opinion about the interaction. A higher opinion means a higher level of satisfaction and vice versa. Social scientists have identified several important factors that are considered while defining the reputation of a member of any group. (1) the opinion in terms of amount of satisfaction a peer receives from another peer, (2) the total number of transactions/interactions a peer has performed, (3) the reputation of the opinion provider reflecting his credibility, (4) temporal adaptability of opinion factor, and (5) the community context factor.

2.3.1 Opinion About An Interaction (O)

Generally, each interaction creates an evaluation about the goodness of a peer. Reputation relies on these individual feedbacks or opinions to evaluate a stable measure about the goodness of a peer. In any collaborative community, a feedback is an indicator of how efficiently and honestly, a peer carried out his side of the interaction. This is the estimate expressed by one member of the community about another. If we consider from an e-commerce perspective, this interaction is nothing but a transaction between two individuals. In many online systems the reputation of a peer is simply, an average or summation of the feedbacks it receives for the various transactions it has been part of. Equation 1 gives a summation and averaging function, which is being used by many pioneering systems such as eBay.

$$R_A = \sum_{j=1}^{N} O_j \qquad (1)$$

In such a system the buyer can leave a positive (+1), a negative (-1) or a neutral (0) feedback. The reputation of the peer is evaluated as the sum of these feedbacks. It is evident that this

system for calculating reputation contains case specific semantics of transaction. Using this equation the reputation of a person who has performed 20 good transactions (reputation = 20) is same as the one who has performed 21 good transactions and 1 bad transaction (reputation = 21 + (-1) = 20), where the two situations are not necessarily identical. Semantics of some transaction may consider the negative to be weighted heavily, which in some other case it might be perfect to just compute a sum. It is further complex because in many cases it has not been possible to establish mapping between a positive score of social mechanism (such as reputation, trust, etc.) with its semantic antonym (such as trust is not necessarily negative of mistrust).

2.3.2 Reputation Of Opinion Provider (R)

Whenever a peer expresses an opinion, many social scenarios seem not to take into account as to who exactly is providing this opinion. They do not make distinction between the opinion providers. For example, in real life one considers the opinion provided by a priest more creditable than the one provided by a thief. Why does one trust the priest more than the thief? This is because the priest has a higher individual reputation in society than the thief. The opinion from those with higher reputation is often weighted more heavily than those with lower reputation. While some systems- such as most voting does not distinguished between individual opinions providers.

2.3.3 Age Of The Opinion (T)

In many scenarios, it seems the age of opinion is often considered an important factor in calculating reputation. By age, what we mean is the freshness of the opinion. Thus an opinion

expressed for a transaction which took place somewhere in the past might have less significance to the opinion which was expressed to a more recent one. A model must have some way to gradually decay the impact of the opinions, as they get older. By incorporating such temporal adaptivity some social systems tends to encourage honest and good peers to remain honest. Due to the aging factor in our system a peer cannot sit on his past laurels and start misbehaving ,because his recent opinions would be the ones which impact his reputation the most rather than the older ones.

2.3.4 Number Of Transactions (N)

As we have mentioned earlier the summation equation is not a reliable indicator of the overall reputation of a peer. In this system, a peer can hide his misbehavior by simple increasing the volume or number of transactions he indulges in. Thus, the total number of transactions is an important factor in determining the reputation of different peers irrespective of the volume of transaction they undertake. A modification to the summation equation (equation 1) can be defined as the ratio of the summation of the different feedback and the total number of transactions. Applying this modified equation to the example discussed in section 3.1 we can see that the reputation of the person who performed 20 good transactions would be 20/20 = 1 and the one who had 21 good transactions to one bad transaction would be (21 -1)/22 =0.90 .Thus, we can see a distinction between the reputations of the peers.

2.3.5 Group Reputation (W)

A peer with a high individual reputation will usually be associated with a group whose members are also highly reputed. However, in cases where a highly reputed peer becomes a member of a group whose members are know to misbehave; group reputation becomes an important factor. In our model, the group reputation, which is an average of the reputation of all the members of a group, would be an indictor of the credibility of the opinion provider. Since the lower group reputation is affecting the good peer, he would have an incentive in encouraging the other members to indulge in honest transactions. This would have a dual effect, firstly the other members might stop misbehaving and secondly the good peer would be rewarded for encouraging other members of his group to be honest.

2.3.6 Impact Parameters

We introduce two types of impact parameters the Impact Variable (X) and the Impact Weight (α). These variables are used to control the direction of influence and the amount of influence the above-mentioned variables would have on the overall reputation of the peer. Table 4 gives the notations for the various impact parameters.

Variable	Impact Variable	Impact Weight
Opinion	$\mathbf{X^O}$	$\boldsymbol{\alpha^O}$
Rating	$\mathbf{X^R}$	$\boldsymbol{\alpha^R}$
Time	$\mathbf{X^T}$	$\boldsymbol{\alpha^T}$
Count	$\mathbf{X^N}$	$\boldsymbol{\alpha^N}$
Time Span	$\mathbf{X^S}$	$\boldsymbol{\alpha^S}$

Group Rating	$\mathbf{X^W}$	$\boldsymbol{\alpha^W}$

Table 4 : Notation For Impact Parameters

Finally we bring all the variables together to form a generic reputation function (equation 2), which encompasses the requirements discussed by us in the previous sections and binds them together into a customizable and consistent formula. We call it a "Generic Reputation Function" (GRF).

$$R_A(t) = \sum_{k=1}^{m} W_k \left[\frac{\sum_{j=1}^{N} R_j^{\alpha^{R} \times X^{R}} \times O_j^{\alpha^{O} \times X^{O}} \times e^{(-\lambda Tj)\alpha^{T} \times X^{T}}}{N^{\alpha^{N} \times X^{N}} + \sum_{j=1}^{m} W^{\alpha^{W} \times X^{W}}} \right] + \Phi \; e^{-\lambda / T_n} \qquad (2)$$

2.4. Discussion About The Generic Reputation Function

Reputation in a society seems to be positively correlated to the variables opinion, individual reputation of opinion provider and freshness of the opinion. Hence the generic reputation function is a product of the three variables as oppose to a simple summation function.We have used an averaging function instead of a summation function since we wanted to restrict the value between 0 and 1 with 0 being the lowest and 1 being the highest. The decrease in the freshness of opinion is a gradual process rather than a sudden one which is better depicted by an exponential function rather than the step decay of a linear function. The age of opinion is thus an exponential function because an exponential function represents the behavior of the freshness of an opinion more accurately. Each factor can affect the reputation evaluation process either positively, negatively or have no impact (zero impact) depending upon the environment in which the function is deployed. The impact variable X controls the influence direction of the various factors. Each factor has its one independent impact variable. As the deployment environment change the influence of each factor may vary. Certain factors may be more aggressively involved in the evaluation

process as compared to others. This behavior can be captured by the impact weight variable **α.** Each factor has its own controlling α and by assigning the appropriate values the impact of certain variables can be made more pronounced as oppose to others.

In this section we present a discussion of how the GRF derived using the RQS (equation. 2) addresses the general concerns faced by present day reputation functions. The summation equation (equation 1) is replaced by an averaging function that calculates the reputation of an individual over a period of time. The opinion credibility issue is taken care off by involving the individual reputation (R) of the opinion provider. The decay of opinions with time is addressed by the exponential part of equation 2 where "λ" is used to define the rate at which the opinions would get older. In our system, the individual starts of with some initial reputation instead of zero. The variable "Φ" is used to assign the initial reputation value and it serves the dual purpose of stabilization.

2.5. Recursive Implementation

For a recursive implementation of the reputation function, we use the following formula. Here the only data to be stored in the database is the previous reputation value and the time of the last opinion.

$$R_n = \frac{\left(R_n^{\alpha^R \times X^R} \times O_n^{\alpha^O \times X^O} \times W_n^{\alpha^W \times X^W}\right) + \left(e^{(-\lambda T_j)} \times R_{n-1}\right)}{1 + e^{-\lambda T_j}} \qquad (3)$$

$$\text{where : } T_j = (T_n - T_{n-1})$$

2.6. Canonical Classes Of The Function

One of the key features of GRF is that it is customizable and dynamic. Depending upon the deployment environment, certain variables would impact the reputation where as others wont' be part of the determination process. There are four primary customizable variables viz. R, T, N and W, thus there are sixteen possible ways to customize them. However, through our experiments we have found that only five of these combinations have corresponding real life examples. Table 5 shows the various applications we have found that could use GRF.

Target ~ Evaluator	R	T	N	W
Book ~ Reader	1	1	1	1
Book ~ Author	1	0	0	1
Movie ~ Viewers	0	1	1	1
Movie ~ Critics	1	1	1	0
Article ~ Reviewer	1	1	1	0
Article ~ Writer	1	0	0	1
Article ~ Journal	1	0	1	0
Article ~ Reader	1	1	1	1
Course Material ~ Student	0	1	1	1
Course Material ~ Preparing Instructor	1	0	0	1
Course Material ~ Other Instructors	1	1	1	1
Protocol ~ Companies	1	1	1	1
Protocol ~ Users	0	1	1	1

Table 5 : Real World Examples

2.6.1 A Fading Memory Averaging Function

$$R_A(t) = \left[\frac{\sum_{j=1}^{N} R_j^{\alpha^{R} \times X^{R}} \times O_j^{\alpha^{O} \times X^{O}} \times e^{(-\lambda T j)\alpha^{T} \times X^{T}} \times W^{\alpha^{W} \times X^{W}}}{\sum_{j=1}^{N} e^{(-\lambda T j)}} \right] + \Phi\, e^{-\lambda / T_n} \qquad (4)$$

In equation 4, $R_A(t)$ denotes the reputation of peer "A" at time "t". R_j is the individual reputation of the peer providing the opinion O_j and T_j is the age of the opinion. The value $\Phi e^{-\lambda / T_n}$ is the normalizing factor for stabilizing the value of the reputation. α and X are the impact variables and "λ" is the decay factor. The formula consists of two parts. The first part is the

average amount of reputation a peer receives for its transactions. This average is different that the usual average since here we have taken into consideration the individual reputation of the opinion provider. In addition to this, we also decay the opinion value, as it gets older with time. The second part is to take care that the reputation of the peer does not decay down to zero with time. If a peer does not indulge in any kind of transactions for a long period, there are no fresh opinions coming in. Hence, due to the decay factor the value would eventually reach zero. In order to protect the reputation function from this situation the reputation value stabilizes itself to "**Φ**". It is due to this constant decay of opinion with time, the function is called fading memory. The function remembers the most recent opinion and exponentially forgets the older ones.

Example: Readers expressing opinions about a book. The individual reputation of the reader matters since we want to weight the opinion expressed by a professor more than the opinion of a casual reader. The time of the opinion matters since a potential buyer would like to know the current reputation of the book as oppose to the past reputation. The number of opinions helps in calculating the average reputation of the book and finally the group reputation matters because of the same arguments put forth in section 2.3.5

2.6.2 A Memory-Less Summation Function

$$R_A(t) = \sum_{j=1}^{N} R_j^{\alpha^{R \times_X R}} \times O_j^{\alpha^{O \times_X O}} \times W^{\alpha^{W \times_X W}} \qquad (5)$$

In this scenario, the target is the product but the evaluator is the producer. This is a memory less summation function because in this scenario the producers express their opinions once. This function evaluates the reputation of a product based on the producer/producers reputation, his/their opinion about the product and if applicable the group reputation of the producers.

Example: Authors expressing opinion about their book .Single or multiple authors can be associated with writing a book. These authors in turn might express an opinion about their book. This is always a one-time process. One does not find situations where the authors keep on changing their opinion about their book. Hence, the reputation of the book is simply a summation of the product of author reputation, author opinion and author group reputation.

2.6.3 A Fading Memory Averaging Function Without Opinion Credibility.

$$R_A(t) = \left[\frac{\sum_{j=1}^{N} R_j^{\alpha^{R \times 0}} \times O_j^{\alpha^{O \times X^O}} \times e^{(-\lambda Tj)\alpha^{T \times X^T}} \times W^{\alpha^{W \times X^W}}}{\sum_{j=1}^{N} e^{(-\lambda Tj)}} \right] + \Phi\, e^{-\lambda / T_n} \quad (6)$$

This is again a fading memory averaging function but here only the opinion matters where as the reputation of the opinion provider does not matter. The reputation of the opinion provider is dropped since this function is deployed in scenarios where the opinion providers fairly have the same reputation. Thus, we set the value of X^R to zero. If at some point we want to differentiate between the opinions, we can use the α^O parameter to vary the impact weight of the opinions.

Example: The Movie ~ Viewer example captures this scenario where the individual reputation of the viewers does not have any impact on the reputation of the movie. Since there are so many viewers and they are almost on the same level as far as reputation goes.

2.6.4 A Fading Memory Averaging Function Without Community Context Factor

$$R_A(t) = \left[\frac{\sum_{j=1}^{N} R_j^{\alpha^{R \times X^R}} \times O_j^{\alpha^{O \times X^O}} \times e^{(-\lambda Tj)\alpha^{T \times X^T}} \times W^{\alpha^{W \times 0}}}{\sum_{j=1}^{N} e^{(-\lambda Tj)}} \right] + \Phi\, e^{-\lambda / T_n} \quad (7)$$

Here we do not include group reputation in the computation of reputation. This is due to two reasons. Firstly, the evaluators cannot be further divided into distinct groups. Secondly, they represent a part of the society that is best in their field.

Example: The example to critics providing opinion about a movie exposes this scenario where the critics cannot be distinguished from each other by grouping them. Thus, since we are not able to form independent groups the community reputation variable does not come into picture.

2.6.5 A Memory-Less Averaging Function

$$R_{A}(t) = \left[\frac{\sum_{j=1}^{N} R_{j}^{\alpha^{R \times X^{R}}} \times O_{j}^{\alpha^{O \times X^{O}}} \times W^{\alpha^{W \times 0}}}{N} \right] \quad (8)$$

In this case, the target is the product and the evaluators are the producers. Here we do not take the group reputation and the time of opinion while computing the reputation of the target. The reason for not including group reputation is the same as that for equation 7 and that for not including time of opinion is same as for equation 5.

2.7. Threats To The Model

Nielson et al ([12]) have identified and created taxonomy for rational attacks and then identified the corresponding solutions if they exist. In their work we see a clear classification of the various types of attacks faced by distributed and peer-to-peer systems. The threats enumerated by Dellarocas ([13]) are similar to the kind of threats that we have tackled.

In the next sections, we have presented a detailed description of the various attacks that are possible on reputation management systems. We have classified the attacks based on set theory as one-one, one-many, many-one and many-many attack. In section 2.8, we introduce the various parties involved in the attack followed by an explanation of the various attacks in section 2.9.

2.8. Parties Involved In Attacks.

(i) Attacker(s) (AT): The attacker/perpetrator can either be the person giving an opinion about the target individual or the target individual himself. We assume that the attacker always lies. The attacker person/group can be of three types. (i) Average Group: It contains a mixture of members having high reputation and low reputation. (ii) Very Good Group: All the members of this group have high reputation. (iii)Very Bad Group: All the members of this group have low reputation. (ii) Evaluators (EV): They represent the general population and are essentially the Controller Conglomeration, which provides random correct opinion about the target individual. We assume in our system that the evaluators never lie and are always truthful. The Evaluator contains members who have a range of reputations from high to low. (iii) Target (TG): The target could be a single individual of a group of individuals. (iv)Offender (OF): The offender is the person who commits something bad in the system for which he should be penalized. He is not an attacker, but he has intentional committed an offense.

2.9. Various Reputation Attacks

	One	**Many**
One	Vendetta	Dr Jekyll & Mr. Hyde

Many	Gang Attack, Praise Planting	Mutual Boosting

Table 6 : Classification Of Attacks

2.9.1 Vendetta

An attacker may target a single user he does not like by giving him a low opinion. This attacker could have High, Low or Average Individual Reputation. The impact of the attack differs depending upon the individual reputation of the attacker

2.9.2 Gang Attack

The attacker can join group of other attacker to reduce the reputation of the target. The attacking group provides unfairly negative opinions to the targeted good user, thereby lowering his reputation.

2.9.3 Praise Planting

The attacker group can increase the reputation of a target by providing unfairly positive opinions to the targeted user, thereby boosting his reputation.

2.9.4 Mutual Boosting

Two-attacker groups join together to mutually inflate their respective reputations by giving each other unfair high opinion.

2.9.5 Dr Jekyll & Mr. Hyde

An offender starts of in the system in a well-behaved manner. As a result, his reputation in the system goes up. Once his reputation is sufficiently high he suddenly turns evil.

2.10. Experimental Evaluation

We performed four sets of experiments to evaluate our Reputation Model. Through these experiments, we prove that out model stands it ground in the face of different attacks. There will not be any kind of attacks on the "Memory Less Summation Function" and the "Fading Memory Averaging Function" since the opinion providers are the producers and they would not want to malign their product's reputation on purpose.

2.11. Vendetta

This scenario involves two individuals where the attacker could be average, good or bad giving a low opinion to the target. The evaluator population is random providing honest opinion to the target.

We have a single target and a single attacker. The evaluator population consists of around 100 peers, which are expressing their honest opinion about their transactions with the target peer. The attacker peer constantly provides false lower opinion to the target peer about their transactions. A summary of the simulation design is given in table 7. The graphs are plotted with final reputation on Y-axis versus the time of the opinion on the X-axis

Type Of Attacker		Opinion Given to the Target
Average	Gives	Low Opinion
Very Good	Gives	Low Opinion
Very Bad	Gives	Low Opinion

Table 7 : Vendetta

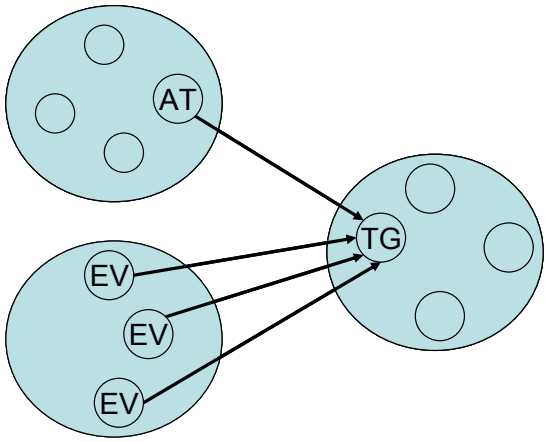


Figure 3: Vendetta

2.11.1 Fading Memory Averaging Function Vendetta Results

The two ellipses in figure 4 denote different periods of attack. In the first ellipse, we can observe that the reputation of the target goes down whereas in the period denoted by the second ellipse the reputation steadily rises even though there are attacks. This contradictory behavior is due to the variable attacker and evaluator opinion frequencies. During the first period, the attacker frequency is higher than the evaluator frequency, due to which the reputation goes down whereas in the latter period the evaluator frequency is higher than that of the attacker, which makes the reputation, go upwards.

Overall from figure 4, 5 and 6 we can deduce that personal attack has a very limited or no damaging effect on the target reputation if the attacker frequency is low but can have a considerable impact in case of higher attacker frequency.

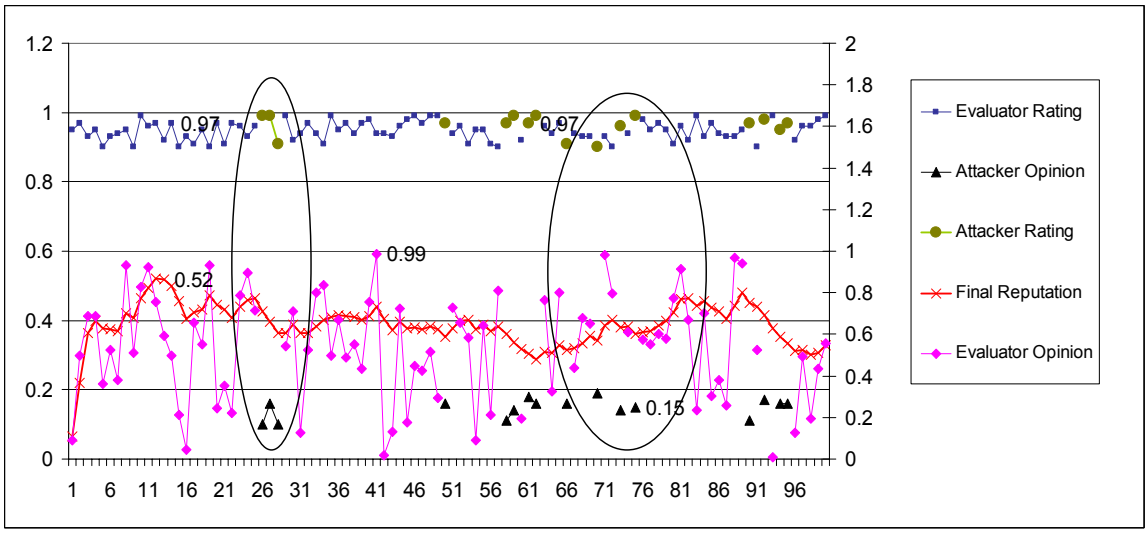


Figure 4 :Behavior of the Reputation Function when the attacker has high personal reputation

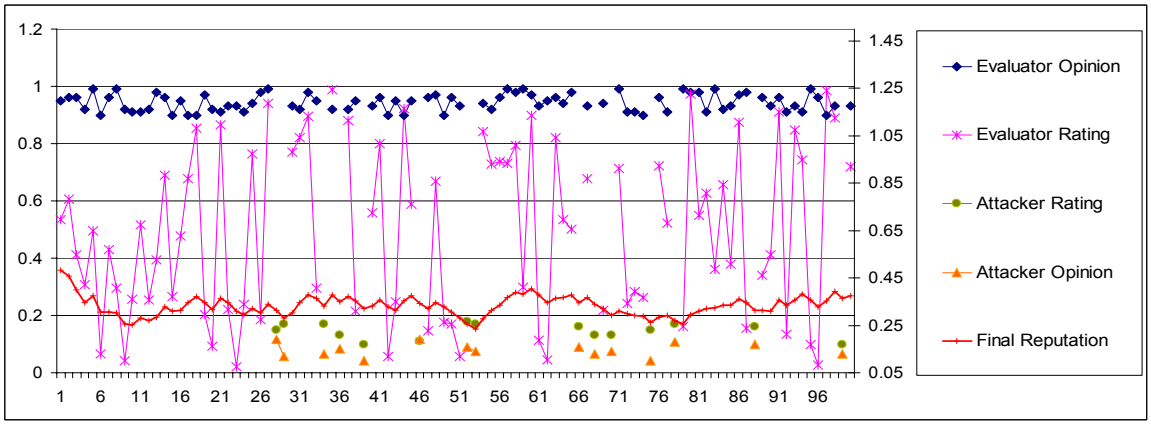


Figure 5: Behavior of the Reputation Function when the attacker has low personal reputation

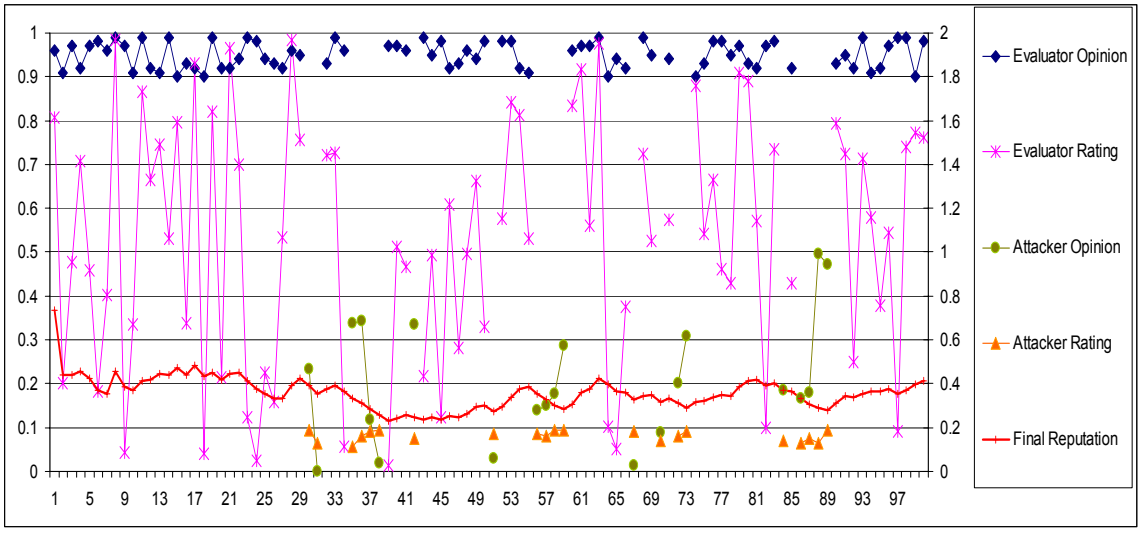


Figure 6: Behavior of the Reputation Function when the attacker has high random reputation

2.11.2 A Fading Memory Averaging Function Without Opinion Credibility Vendetta Results

The results for this function show a behavior similar to the one shown by the fading memory averaging function.

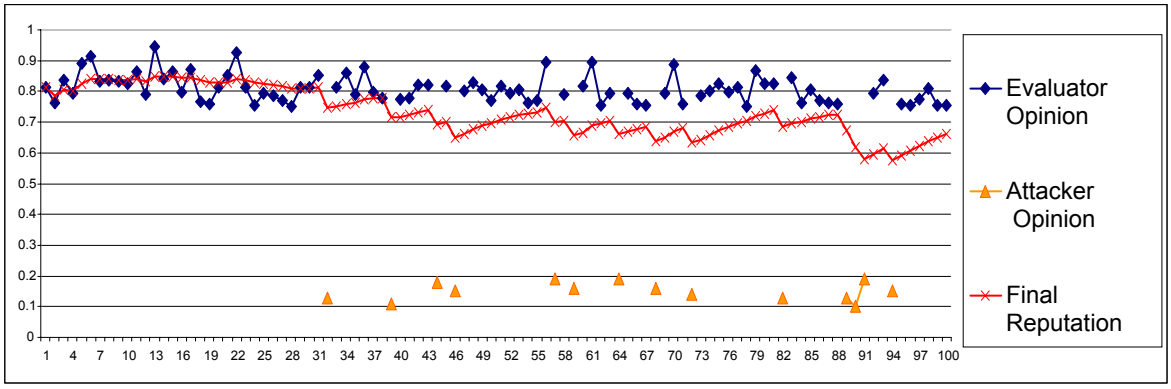


Figure 7 : Behavior of the reputation function without opinion credibility during vendetta

2.11.3 A Fading Memory Averaging Function Without Community Context Factor Vendetta Results

The results for this function show a behavior similar to the one shown by the fading memory averaging function.

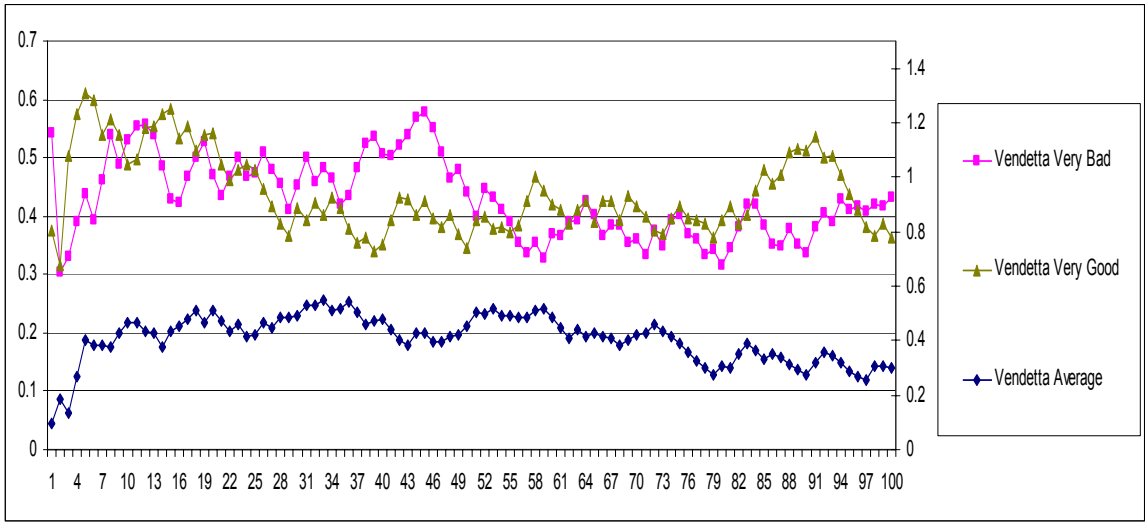


Figure 8 : Behavior of the reputation function for various types of vendetta

2.12. Damaging Gang Attack

This scenario involves a group of attackers turning hostile towards the target peer. This attack is different that Personal attack. In personal attack, the attacker never stops expressing bad opinion about the target whereas in damaging gang attack, the attacking group expresses bad opinion about the target in a short span of time with the intension of pulling down the target's reputation. During the attack period, the frequency at which the attacker group expresses its opinion is higher than the honest evaluator group frequency.

We have a single target and a group of attackers. The attackers are initially part of the evaluator group but abruptly turn evil. The number of members of the attacker group is set to 10 % of the total number of evaluator peer. The graphs are plotted with final reputation on Y-axis versus the time of the opinion on the X-axis. The simulation design is summarized in table 8.

Type Of Attacker Group		Opinion Given
Average	Gives	Low Opinion
Very Good	Gives	Low Opinion
Very Bad	Gives	Low Opinion

Table 8 : Damaging Gang Attack

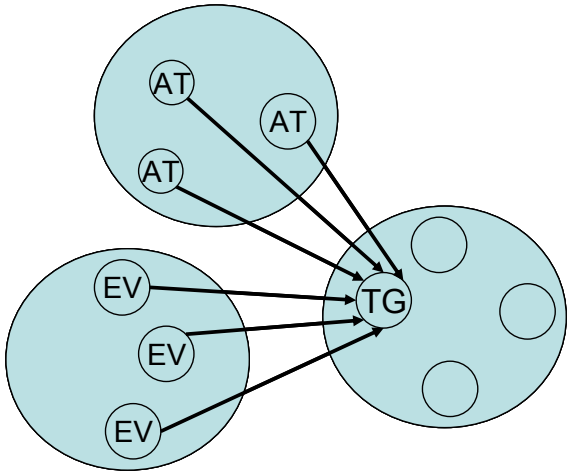


Figure 9 : Damaging Gang Attack

2.12.1 Fading Memory Averaging Function

The rectangular dotted region represents the attack periods. Figures 10, 11 and 12 represent three flavors of the same attack. In figure 10, we can notice two small recoveries of the target's reputation; one of them is represented by the dotted circle. Figure 10 represents a brutal at-

tack where the attacker frequency is five times higher than the evaluator frequency and the reputation just plunges. In figure 12, we see a saw tooth like behavior because the evaluator frequency is only a few times lower than the attacker frequency and hence there are recoveries at regular intervals but eventually the reputation go down.

Through figures 10, 11 and 12 we observe that though the attackers manage to bring down the reputation of the target during the attack period, they are not able to inflict permanent damage. The function recovers itself to the original value through the honest opinion expressed by evaluators with high reputation and the age of the opinion variable.

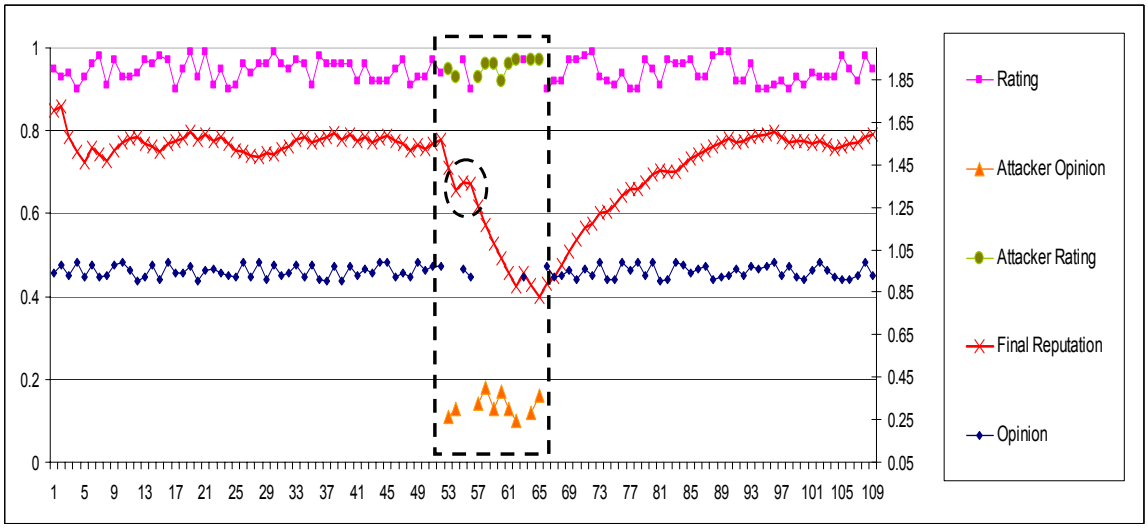


Figure 10 : Behavior of the reputation function when the members of the attacker groupd have high personal reputation

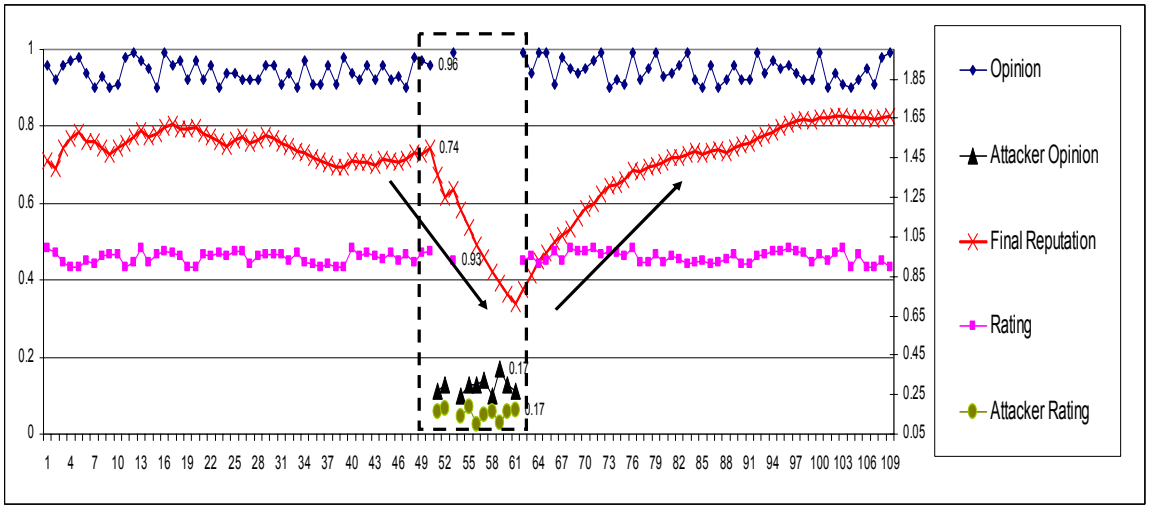


Figure 11 : Behavior of the reputation function when the members of the attacker group have low personal reputation

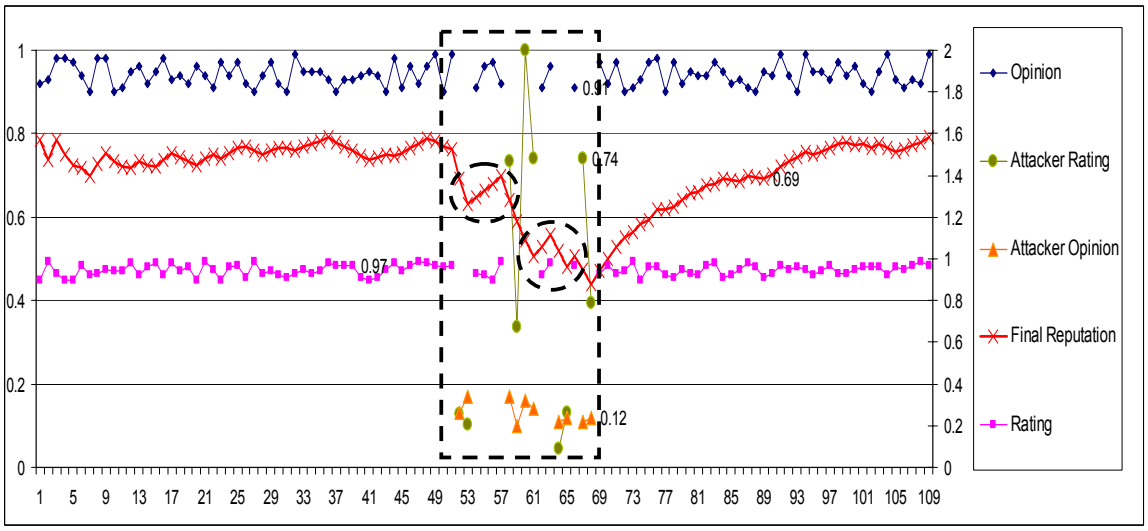


Figure 12 : Behavior of the reputation function when the members of the attacker group have random personal reputation

2.12.2 A Fading Memory Averaging Function Without Opinion Credibility

We observe a behavior similar to the one discussed in section 2.12.1. The reputation goes down during the attack period but then the recovery starts as soon as the attack is over.

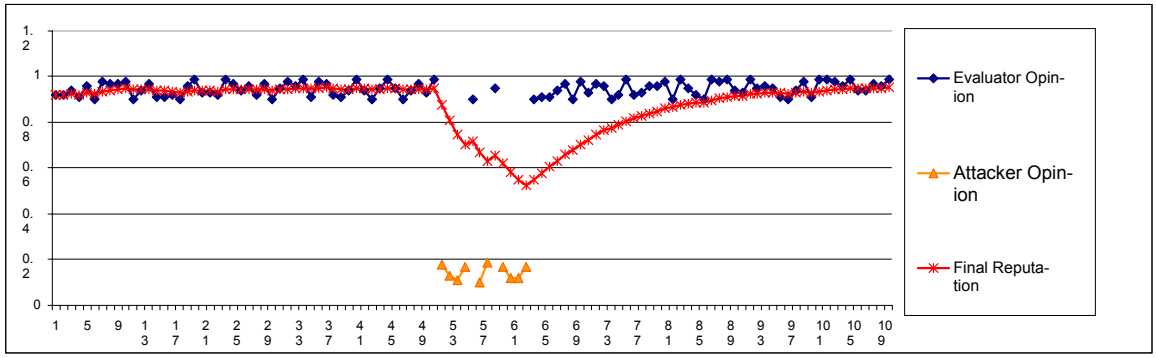


Figure 13 : Behavior of reputation function without opinion credibility under damaging gang attack

2.12.3 A Fading Memory Function Without Community Context Factor

The function results are similar to the fading memory averaging function discussed in section 2.12.1.

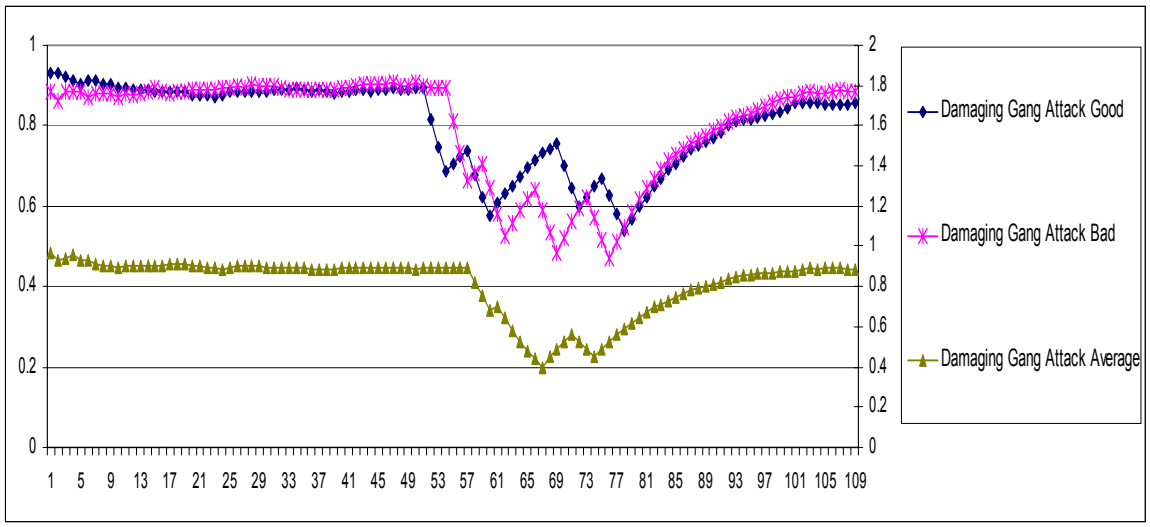


Figure 14 : Behavior of the reputation function without community context factor for various types of damaging gang attacks

2.13. Praise Planting

In this scenario, the friends of a peer (attackers) provide him with false high opinions. This case occurs when a peer has a low reputation and wants to increase his reputation through unjust means. It is similar to the damaging gang attack but here the target is actually a part of the attacking group.

We have a single target and a group of attackers. The attackers are initially part of the evaluator group but abruptly start expressing high opinion for the target. The number of members of the attacker group is set to 10 % of the total number of evaluator peers. The simulation design is summarized in table 9. The graphs are plotted with final reputation on Y-axis versus the time of the opinion on the X-axis.

Attacker Group Type		Opinion Given by Attacker Group
Average	Gives	High Opinion
Very Good	Gives	High Opinion
Very Bad	Gives	High Opinion

Table 9 : Praise Planting

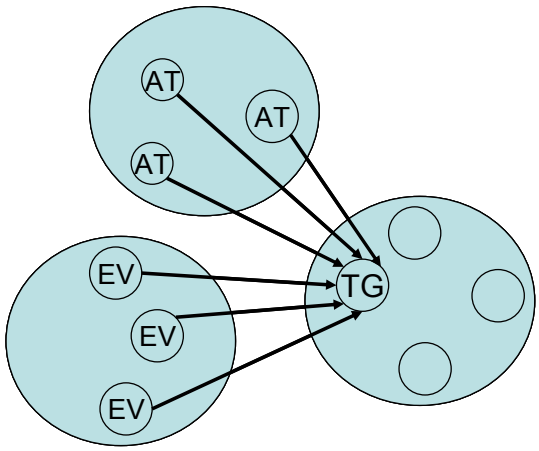


Figure 15 : Praise Planting

2.13.1 Fading Memory Averaging Function

The dotted rectangles represent the attack periods. Figure 16 and figure 18 show a predictable pattern in which the high opinion expressed by the friend group boost the targets reputation. However, figure 17 displays an anomaly to this behavior where the boosting is not helping the target reputation to move up but is actually pulling it down. In this particular scenario, the members of the friend group have low individual reputation. Thus, even though they are provid-

ing high opinion they are having a negative impact and are not able to serve their purpose. The results for this attack are similar to those observed for Damaging Gang Attack. The boosting helps the target to gain reputation for a short amount of time, but the evaluators duly bring it back down. Once the attack period is over the target reputation stabilizes to its original low value.

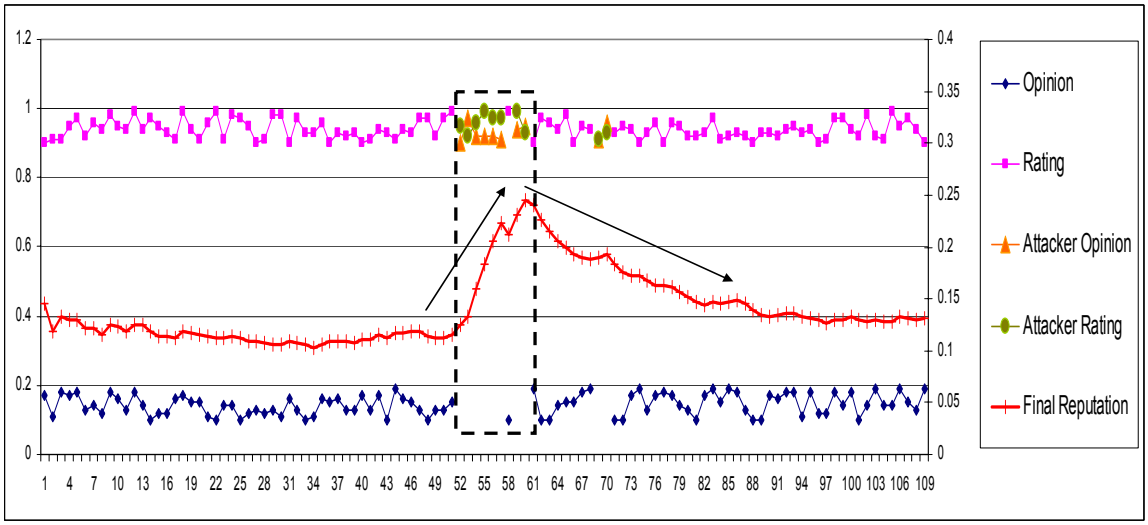


Figure 16 : Behavior of the reputation function when the members of the attacker group have high personal reputation

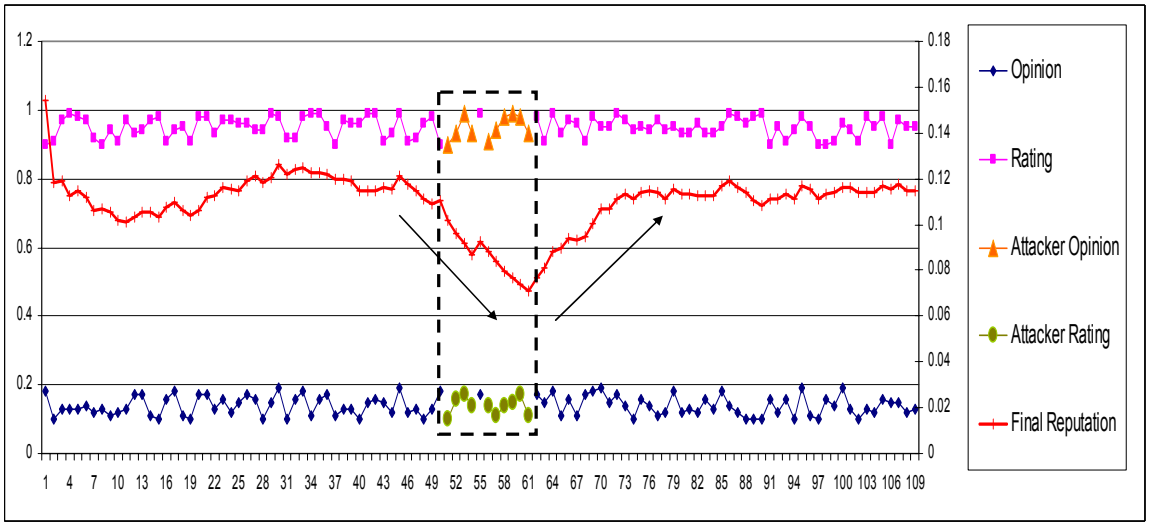


Figure 17 : Behavior of the reputation function when the members of the attacker group have low personal reputation

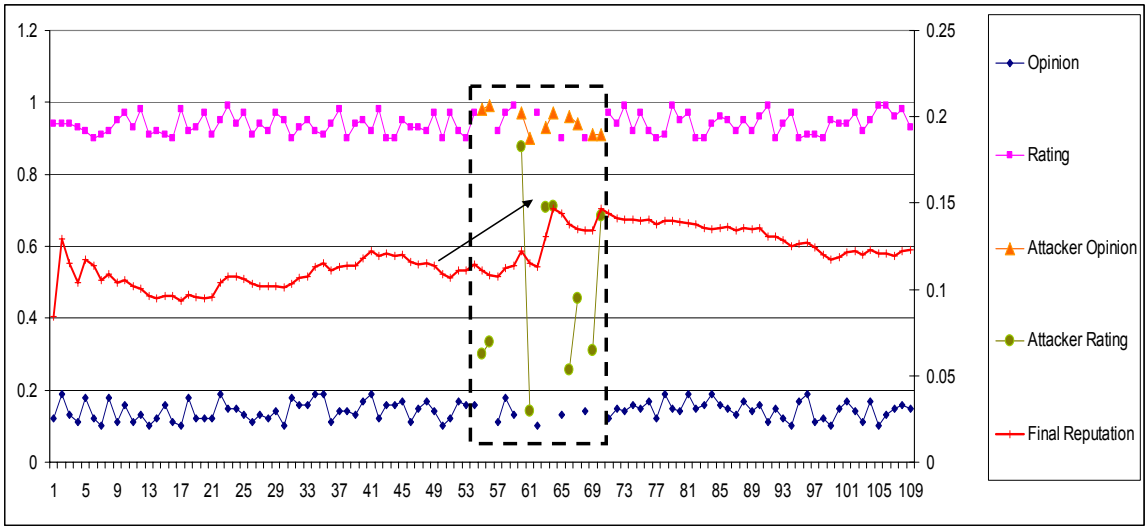


Figure 18 : Behavior of the reputation function when the members of the attacker group have random personal reputation

2.13.2 A Fading Memory Averaging Function Without Opinion Credibility

The results observed are similar to the fading memory averaging function results.

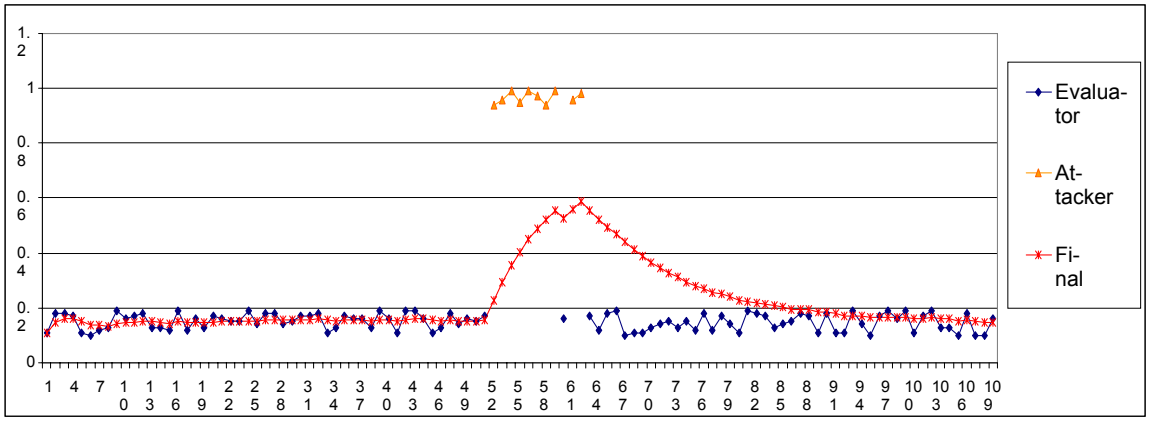


Figure 19 : Behavior of the reputation function without opinion credibility during praise planting

2.13.3 A Fading Memory Averaging Function without Community Context Factor

The results observed are similar to the fading memory averaging function results.

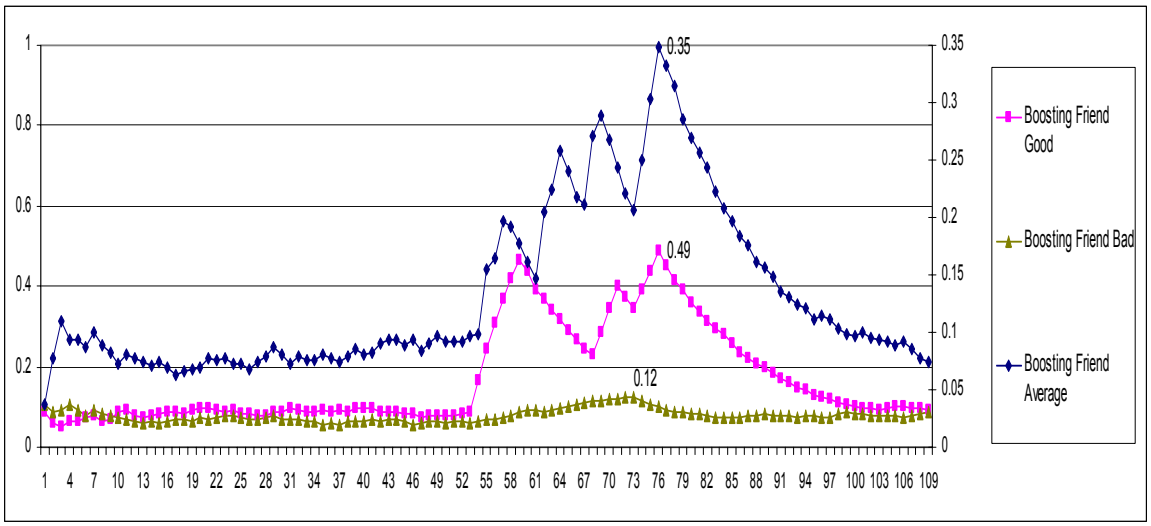


Figure 20 : Behavior of the reputation function without community context factor for various types of praise planting

2.14. Dr Jekyll & Mr. Hyde

This scenario tries to depict the dual conflicting behavior of a peer in the community. Similar to the last scenario the target here is actually the attacker. After developing a high reputation in the system by indulging in honest transactions, the target takes advantage of his high reputation to misbehave. The evaluators penalize the target by giving him low opinions. After penalization, the target reverts to being honest for sometime. Once he has been successful in getting his reputation back to the original high value he starts misbehaving again.

The simulation consists of three groups. The evaluator group in this scenario consists of the average group, the very good group and the very bad group. The simulation design is summarized in the table 10: The graphs are plotted with final reputation on Y-axis versus the time of the opinion on the X-axis

Behavior of Evaluators after Offence		
Reacting Evaluators		Opinion Given by Reacting Evaluators
All Average, Very Good, Very Bad	Gives	Low Opinion
Only Very Good	Gives	Low Opinion

Table 10 : Dr Jekyll & Mr. Hyde

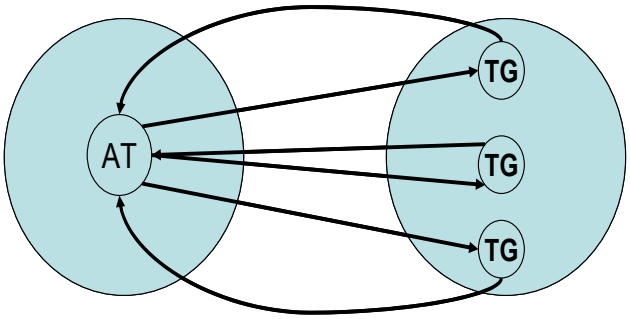


Figure 21: Dr Jekyll & Mr. Hyde

2.14.1 Fading Memory Averaging Function

The Dr Jekyll and Mr. Hyde phenomenon is vividly seen in the figures 22 and 23. The evaluators punish the target for his offence, which results in his reputation taking a downward slide. However, he recovers his reputation through indulging in honest transaction, again to commit offence for which he is duly penalized. This trend is seen in figure 22 by the upward and downward movement of the reputation function

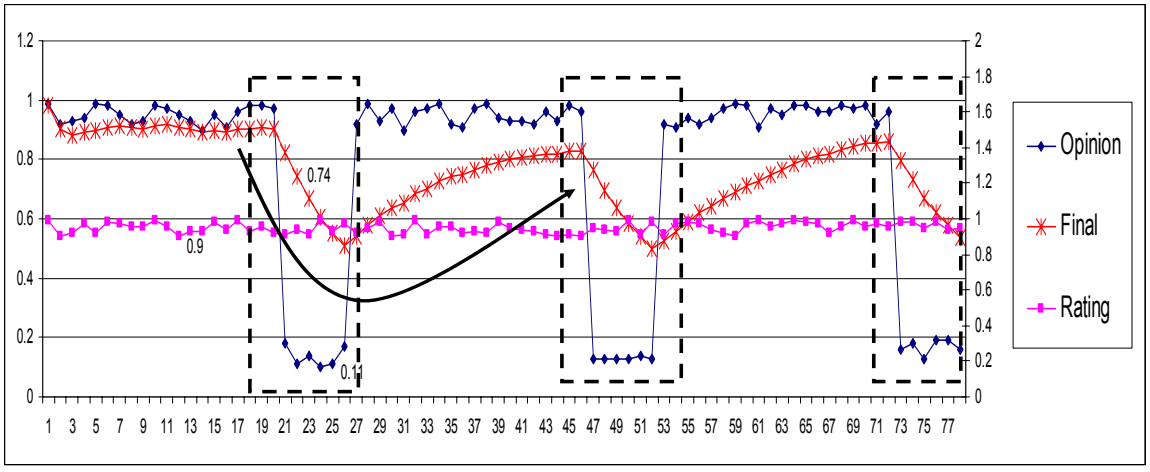


Figure 22 : Behavior of the reputation function when the members of the evaluator group have high personal reputation

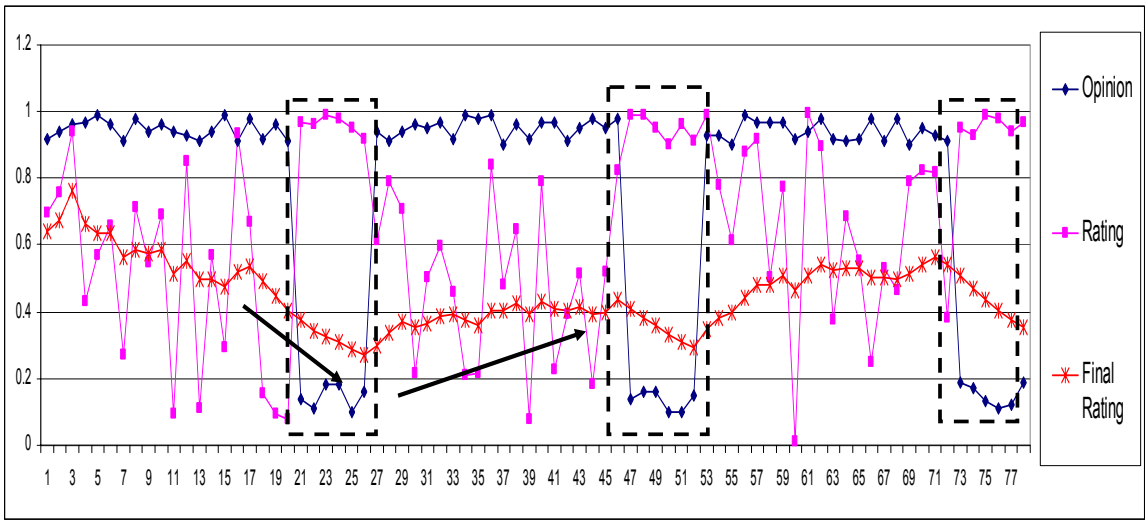


Figure 23 : Behavior of the reputation function when the members of the evaluator group have random personal reputation

2.14.2 A Fading Memory Averaging Function Without Opinion Credibility

The results observed are similar to the fading memory averaging function results.

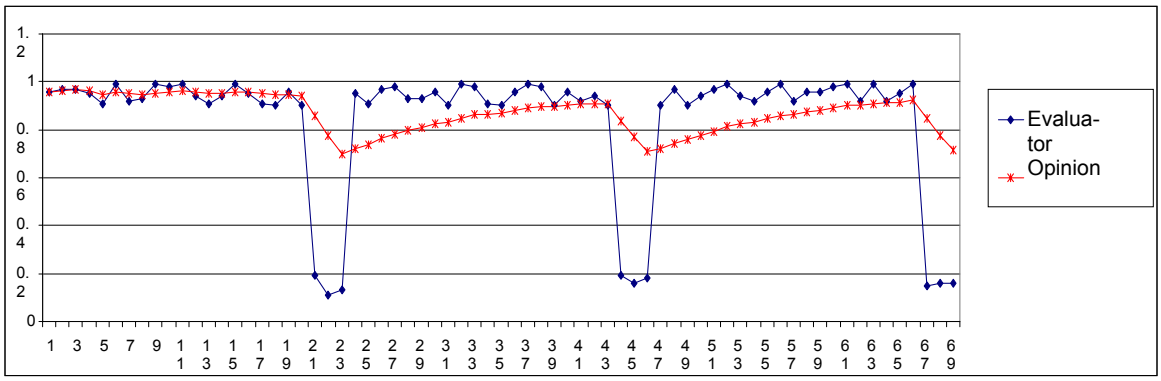


Figure 24 : Behavior of the reputation function without opinion credibility during Dr Jekyll & Mr. Hyde

2.14.3 A Fading Memory Averaging Function Without Community Context Factor

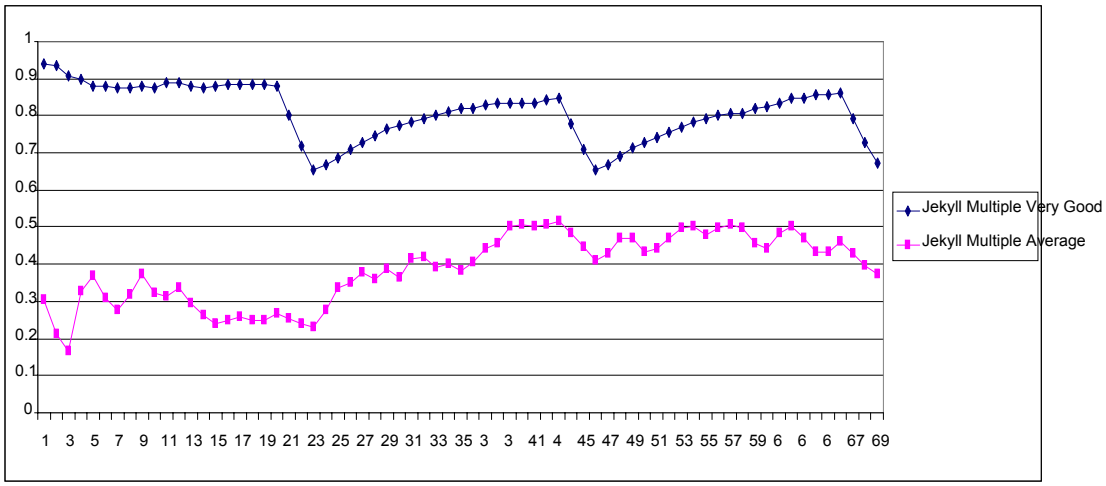


Figure 25 : Behaviors of the reputation function without community context factor for Dr. Jekyll & Mr. Hyde kind of attack

2.15. Mutual Boosting By Groups

In this scenario, small groups of people within the community collaborate with each other to boost their reputations mutually. This coterie of people does this by giving false high opinion. Mutual boosting is different from boosting by friends. Since here, the opinion giver gets an opinion in return for his help.

The simulation consists of two groups the evaluators and the attackers. The simulation design is summarized in the table 11.

Friend Group Type		Opinion Given by the Friend Group
Average	Gives	High Opinion
Very Good	Gives	High Opinion
Very Bad	Gives	High Opinion

Table 11 : Mutual Boosting

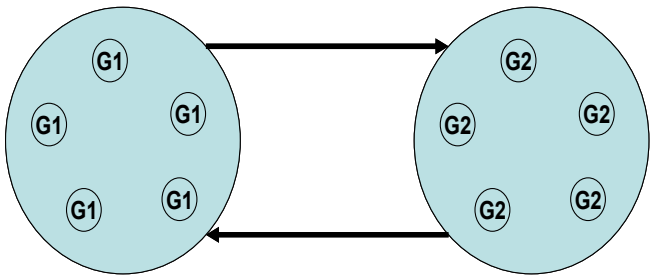


Figure 26 : Mutual Boosting

2.15.1 Fading Memory Averaging Function

The simulation results show that after the boosting is complete all the members of the coterie have almost the same reputation. The coterie would have a mixed bag of people. Some with high individual reputation while others with low and average reputations. The conclusion we can draw from this result is that since the high reputation individual expresses a false opinion about the lowly rated peer, he loses his standing in the community and his reputation drops. The only people gaining from this are the ones with low opinions. Their reputation increases a notch and at the end of the attack, members of the clique have almost the same reputation.

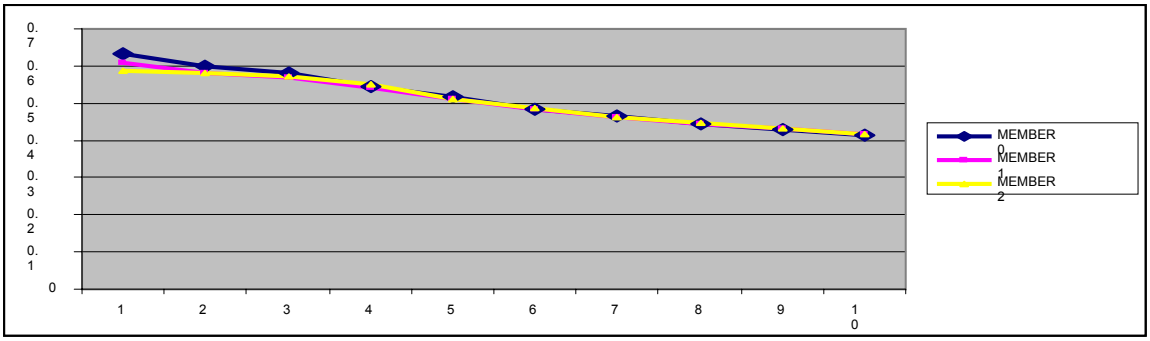


Figure 27 : Behavior of the reputation function

2.15.2 A Fading Memory Averaging Function Without Community Context Factor

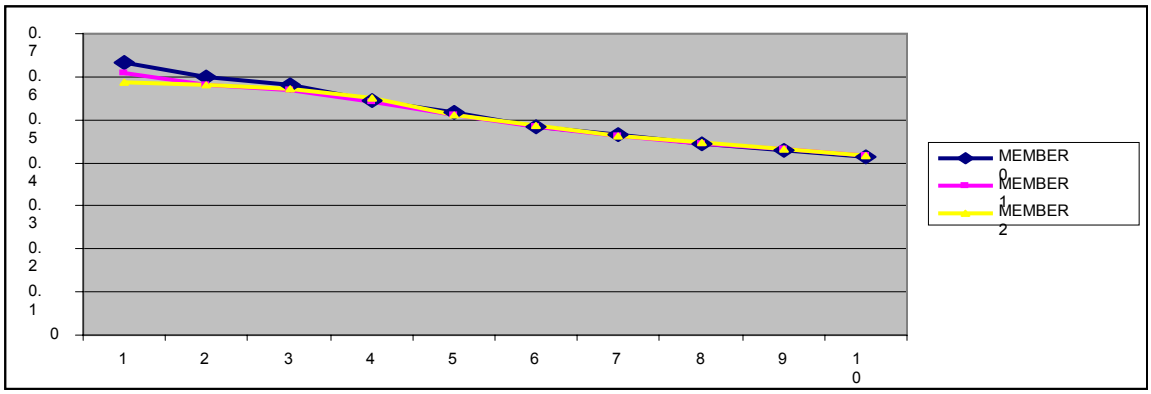


Figure 28 : Reputations of the members of the mutual boosting clique

2.16. A Memory Less Averaging Function

There will not be any kind of attacks on case 5 scenarios since the opinion providers are affiliated to the producers and they would not want to malign their product reputation on purpose. The graph below shows how individual reputation of the producers affects the overall reputation of the product. The results are straightforward, as they indicate that the product reputation is directly proportional to the producer reputation.

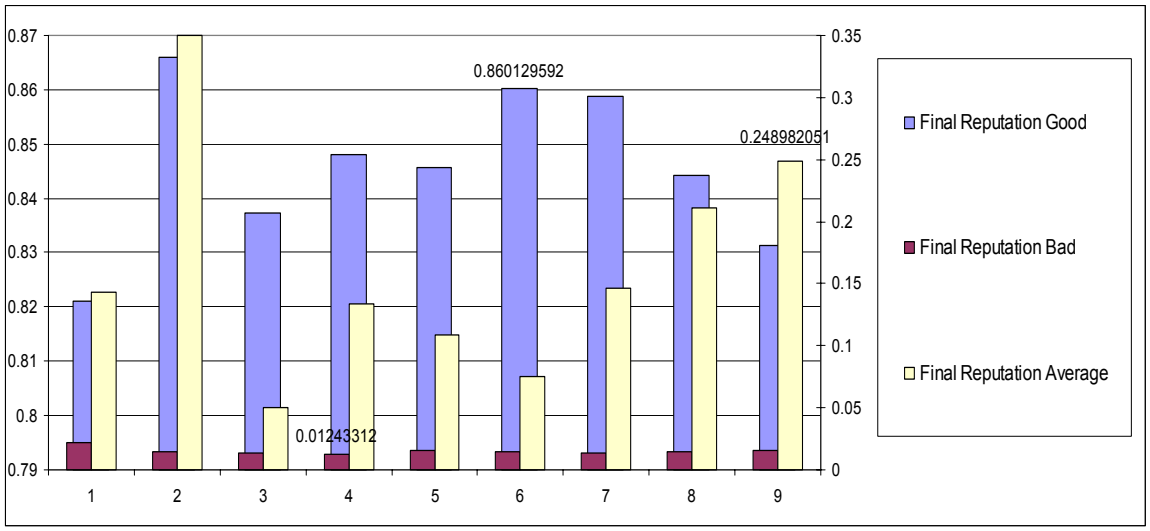


Figure 29 : Reputation of the product for different number and different types of producers

2.17. Conclusion

We presented a generic reputation function, which can be customized to be used in various different environments. We identified the core factors that can affect the reputation of an individual. In most of the other reputation functions, the core factors are static whereas in our function they can be changed according to the demands of the environment. Thus, we have a single function, which can serve in an ecommerce website or any online group activity or in a peer-to-peer system by just tweaking a few variables here and there. We have also provided experimental results to prove that are function is robust and effective against various reputation attacks.

Through our experiments, we have observed that there is no definite way of distinguishing between Damaging gang and Dr. Jekyll & Mr. Hide kind of attacks. A method, which could give some kind of indication as to which attack is in progress, is the number of low opinions being expressed towards the target. If the percentage of these opinions is between 1 – 5 % then we can say that it is a damaging gang attack. However, if the percentage is in between 10 – 30 % we

can conclude that it is a Dr. Jekyll and Mr. Hide scenario and the evaluators are penalizing the target for some offence he committed.

CHAPTER 3

SOCIAL NETWORK BASED COMPUTATIONS

We have proposed a plausible classification of social network based computations on the underlying algorithmic structure that resemble the classical algorithms but have been modified to complement social computations. They have been classified as social profile mining, social fabric analysis, social linkage analysis, social ranking analysis and social placement analysis. A graphical representation of the various applications is given by Figure 30. The following sections present each class of computation and their interesting practical applications.

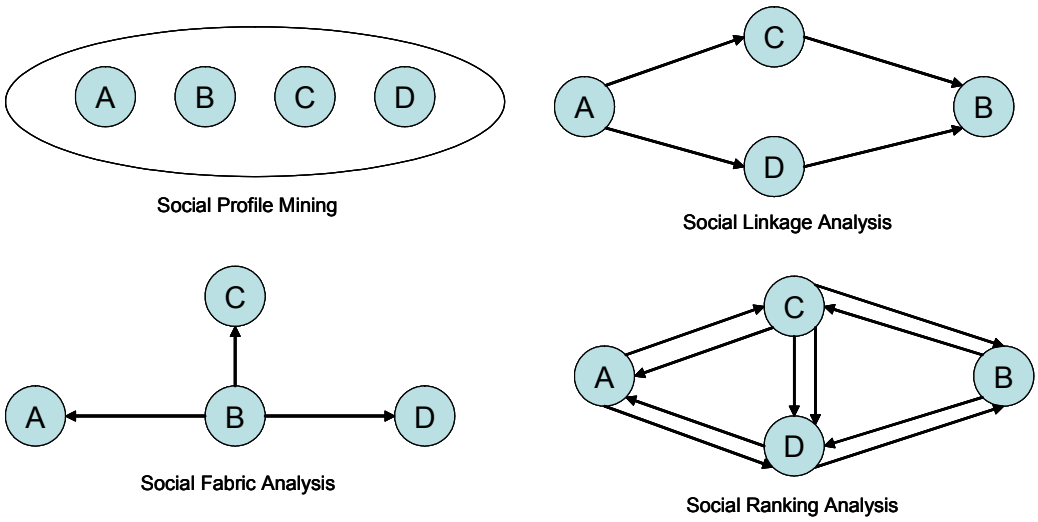


Figure 30 : Social Network Based Computations Classification

3.1 Social Profile Mining

Social profile mining is the process of searching large volumes of data, collected from social networking websites, for hidden patterns that can be used to estimate future behavior. The applications of this class of computation are pure statistical analysis of a database. The computations can be used for customer geo-profiling, demographic analysis and for comparative market analysis.

The target database can be obtained from the increasing popular social networking websites. These websites gather information through a large questionnaire, which has to be filled by any user who wishes to use their applications. The questionnaire is targeted towards profiling the user on the basis of his/her personal and professional information. Figure xx gives a snapshot view of the questionnaire. Since, these websites are used for social interactions, one can optimistically assume that the information being shared by users is truthful and fairly precise. In this process, a rich database is being created, which can be used by data miners for a number of application one of which could be to estimate the current trends and behavior of a community.

3.1.1 Various Statistical Analysis

The data collected can be subjected to statistical analysis (Journal Statistical Analysis and Data Mining). The analysis could be descriptive or inferential (SPSS). Descriptive statistics can be used to summarize the data using numerical descriptors such as mean, mode and standard deviation. Inferential statistics is used to model patterns in the data, accounting for randomness and drawing inferences about the larger population. These inferences may take the form of answers to yes/no questions (hypothesis testing), estimates of numerical characteristics (estimation), forecasting of future observations, descriptions of association (correlation), or modeling of relationships (regression). Some well know statistical tests and procedures such as t-test, chi-square test,

analysis of variance, correlation, regression analysis and cross tabulation can be applied on the data. In the next section, we give an example of some primary statistical analysis performed on data collected from a popular social networking website

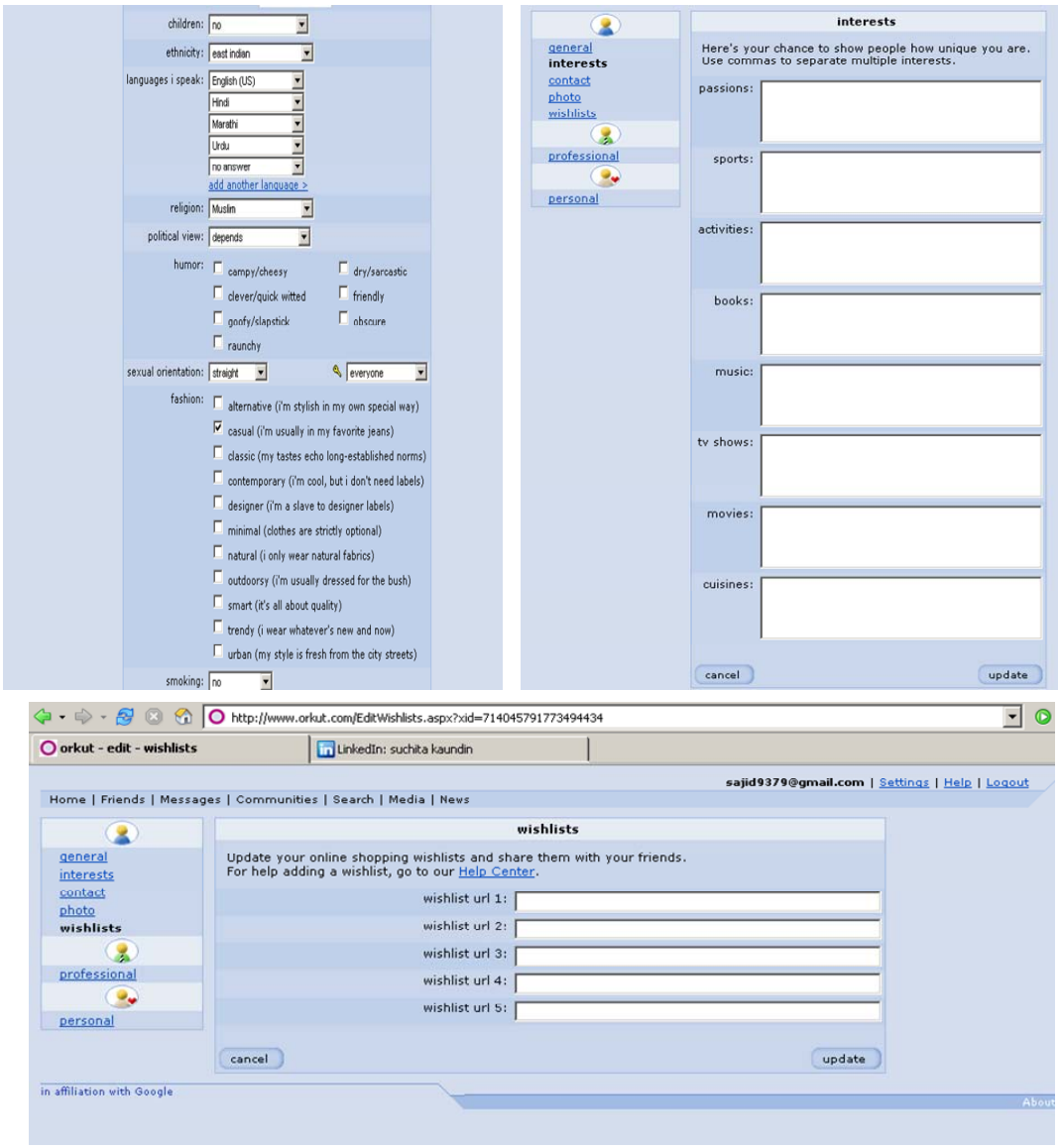


Figure 31: An Orkut Profile Questionnaire

3.1.2 Examples

The examples being presented in this section, use data collected from Orkut (http://www.orkut.com), a social networking service provided by Google ©. The examples use information about Orkut users in Austin, Texas and Kent, Ohio.

The first example is aimed at finding the culinary preferences of Austin residents. This can be fond out just by mining the cuisine section of profiles of various Orkut members in and around Austin. The results of such a data mining carried out for 100 random residents indicate that the most popular cuisines are Indian (68%), Italian (46%), Mexican (43%), Chinese (37%), Thai (32%) and Japanese (19%).

%	Indian	Italian	Mexican	Chinese	Thai	Japanese
Indian		27.6	29.6	26.5	26.5	6.1
Italian	27.6		22.4	17.3	15.3	12.2
Mexican	29.6	22.4		18.4	16.3	10.2
Chinese	26.5	17.3	18.4		10.2	9.2
Thai	26.5	15.3	16.3	10.2		5.1
Japanese	6.1	12.2	10.2	9.2	5.1	

Table 12 : Austin Resident's Cuisine Data Analysis

The second example is aimed at estimating the kind to movies people around Kent like to watch. The results of the data mining were Drama (76%), Action (63%), Thriller (60%), Crime (59%), and Adventure (49%). This information can be used by the local movie theatre to choose what kind of movies would bring in more viewers and increase the theatre's revenue. Cross-tabulation analysis on the collected data gave the following results.

%	Drama	Action	Thriller	Crime	Adventure
Drama		51.2	53.7	53.7	34.1
Action	51.2		43.9	31.7	41.5
Thriller	53.7	43.9		43.9	26.8
Crime	53.7	31.7	43.9		19.5
Adventure	34.1	42.5	26.8	19.5	

Table 13 : Resident's Movie Preference Data Analysis

A number of popularity-based applications can be implemented using this database. Such as, what kind of music is popular? Which websites are most popular? Which TV shows are popular? Which sports are popular? Etc.

3.1.3 Discussion

In addition to statistical analysis, advanced data mining techniques beyond statistical analysis could be employed as well. One can use classical data mining algorithms such as clustering and nearest neighbor prediction techniques. Alternatively, one could use newer techniques such as decision trees, neural networks and rule induction for discovering new information within large databases or for building predictive models.

3.2 Social Fabric Analysis

Social fabric analysis deals with the derivation of social properties in an individual's social network using primary relationship chains. In social fabric analysis based applications, the focus is on a single individual and the part of the neighborhood created by hi/her primary relationships. The computation is single-individual centric and tracks the individual's relationships graph to assess various social properties.

While creating the social network of an individual we are considering the following relationship domains namely, friends, relatives, coworkers and enemies. Some of the relationships in these domains are direct or primary relationships, which are used to derive the other indirect or secondary relationships. The primary and the secondary relationships together are used to deduce complex social properties such as influence, trust, status etc. Since a social network can

be an enormous graph, the computation are limited to only that part of the graph which depicts the individuals primary relationships and certain secondary relationships. The relationship graph is pruned and the depth of traversal limited by applying certain social factors which act as constraints. These constraints change according to the social property one is trying to derive. The next section gives an overview about one such social property, influence, by quoting a classical definition and providing a discussion about the factors that affect influence. In section 3.2.2 we present an algorithmic structure for determining influence and we show its application by working out an example in section 3.2.3.

3.2.1 Example: Influence Assessment

Miriam-Webster dictionary defines influence as "the act or power of producing an effect without apparent exertion of force or direct exercise of command" or "to affect or alter by indirect or intangible means". In chapter 12 of Canadian Organization Behavior, the authors present various types of influence, which are Silent Authority, Assertiveness, Exchange, Coalition Formation, Upward Appeal, Persuasion and Information Control.

Social science literature identifies a number of factors that affect influence, but we consider the ones, which we think, are the most dominant. They are called I-factors. The I-factors are of different types, a few are individual's personal properties such as age, location while others depend upon the interactions between individual and his social circle, such as frequency of contact and trust. A few I-factors depend upon the relationship network, such as type of relationship while others depend upon the behavior of individuals in society such as reputation. In the following sections, we give a detailed discussion of the I-factors.

(i) Age difference (F1): An elderly person is more likely to be influential than a younger person. (ii) Proximity of the nodes (F2): If two persons have the same degree of influence upon the individual but one of them is physically closer, then the influencing power of the closer person is more likely to increase. (iii) Type of relationship (F3): In most likelihood, the primary relationships would have more influence than the secondary ones. (iv) Frequency of contact (F4): A person whom the individual meets regularly has a higher probability of influencing him/her than a person whom he/she does not meet that often. (v) Context (F5): For example if the individual wants to buy a car, then his/her friend who is a car expert will have more influence on his/her decision than say his/her spouse, who does not know much about cars. Therefore, even though he/she has a stronger relationship with his/her spouse than he/she does with a friend, the context increases the influencing power of his/her friend over him/her. (vi) Reputation of the influencing node (F6): A highly reputed person in the individual's social network will have more influence on him than a person with a low reputation. (vii) Trust (F7): A person the individual trusts would have influence on him rather than a person he distrust.

While evaluating influence in different environments, it is possible that not all the factors enumerated above would be needed. For example, if we consider the factors in context with a church then the priest has influence on a variety of people who visit the church. Even if the priest is younger than the worshippers are, it does not have an effect on his influencing power. Thus, in this case we have to disregard the age I-factor. In another scenario, suppose the frequency of contact between an individual and his/her best friend is not that high due to geographical distance. Nevertheless, that does not diminish the influencing power each one has over the other. In this scenario, the frequency of contact I-factor is ignored.

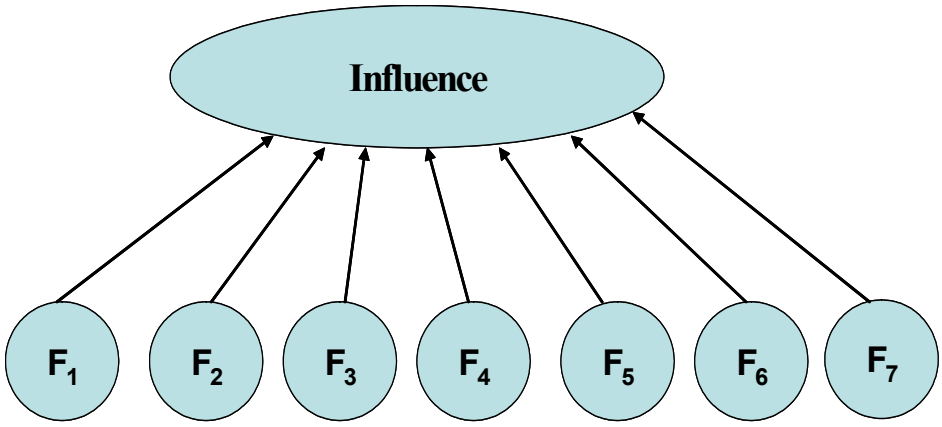


Figure 32 : Influence Assessment

The aim is to formulate all the above-mentioned factors into a consistent function.

The age of each individual can be determined from his profile on Orkut. Thus, age difference F_1 is given by equation 9.

$$F_1(i,j) = Age_i - Age_j \tag{9}$$

Proximity of individuals can be determined by location field in Orkut. Thus, the factor F_2 is given by equation 10.

$$F_2(i,j) = f\left[city(i,j), state(i,j), country(i.j)\right] \tag{10}$$

Type of relationship can be determined from the social network and by monitoring the interaction between the individuals. These interactions could be through e-mails, blogs, scarping, chatting etc. Thus, F_3 is given by equation 11.

$$F_3(i,j) = relation(i,j) \tag{11}$$

The frequency of contact can be calculated form the number of scraps that occur over a period. The time and date of the scraps are available for public view. Factor F_4 is given by equation 12.

$$F_4(i,j) = \frac{\sum_{k=1}^{N} scrap\ (i,j)}{N} \tag{12}$$

The reputation of the individual can be determined using the opinion expressed about the individual by his social network. Equation 13 gives a formula for calculating individual reputation.

$$F_{6}(A) = \left[\frac{\sum_{j=1}^{N} R_{j}^{\alpha R \times X R} O_{j}^{\alpha O \times X O} e^{(-\lambda Tj)^{\alpha T \times X T}}}{\sum_{j=1}^{N} e^{(-\lambda Tj)}} \right] + \Phi\, e^{-\lambda / Tn} \tag{13}$$

We can quantify the trust relationship by using the karma rating (smiley faces in Orkut) and plugging them into the "Fading Memory Averaging Function".

$$F_{7}(i,j) = R_{i}^{\alpha R \times X R} \left[\frac{\sum_{m=1}^{N} K_{m}^{\alpha O \times X O} \times e^{(-\lambda Tj)^{\alpha T \times X T}}}{\sum_{m=1}^{N} e^{(-\lambda Tm)}} \right] + \Phi\, e^{-\lambda / Tn} \tag{14}$$

Equation 14 denotes the amount of trust node "i" has in node "j". R_i is the individual reputation of peer "i" and K_m is the trust opinions that "i" express about "j". T_m denotes the age of the opinion. The second part of the equation is the normalizing factor for stabilizing trust value. α and X are the impact variables and "λ" is the decay factor.

Thus, the complete mathematical formulation for influence is given by equation 15.

$$I = f(F_1, F_2, F_3, F_4, F_5, F_6, F_7) \tag{15}$$

3.2.2 Algorithmic Sketch For Determining Influence

The proposed algorithmic sketch is based on a depth-first traversal of the social network with the individual as the root. It starts at the root and depending upon the relationship link decides whether to visit the next node or not. The selection of which link to follow is determined based upon a table containing a list of favorable relationship chains. Once it reaches the next node, it evaluates the node based on a list of constraints. If the node satisfies those constraints then it continues a depth-first traversal else, it retracts back. The depth of traversal is also depend-

ant upon the relationship chain table. The table contains a list and the length up to which a particular relationship chain should be followed.

The crux of the algorithmic sketch is to follow those links, which will increase the influencing value while there are others, which should be avoided. The people who are most likely of influencing an individual are within 2 hops in his/her social network. They consist of friend (F), father (FA), mother (MO), son (SO), daughter (DA) spouse (SP), grandfather (GF), grandmother (GM). As the hops, increase the influencing power tends to decrease. Thus some of the influential chains that would yield favorable results are: F{1,2}, F(FA|MO|GF|GM), SO[F{1,2}], SO{1,2}, SP(FA|MO).

However, in some cases, the influencing power would not decrease as it keeps getting longer but it might remain more or less the same. This behavior is seen in a chain of boss (BO) and subordinate (SU) relationship: (BO|SU){1,}

Each of these links can be given different weight depending upon their influencing power. Thus, while traversing an individual's social network when one comes across these chains then one can infer that one is along the right path for influence.

```
Influence (G, u, δ, λ)    u - is the source node
        Stack S = { };
        Stack O = { };
        Boolean x, y;
        pathString (u) = null;
        Push S, u;
        while ( S is not empty ) do
            u := Pop S;
            Push S, u;
            for each vertex v adjacent to u
              if ( Age_V > 18 ) && ( F_1(u, v) > 2) && ( F_6(v) > δ) && (F_7(u, v) > λ )
              I = (Age_V-/100) * (F_1 (u, v)/50) * F_6 (v) * F_7 (u, v);
              pathString(v) = Concatenate ( pathString(u), F_3(u,v));
              Push S, v;
              StringMatching(pathString(v), patternDb())
              I = I * patternDb(pathString(v)).value();
              return I;
              end if
        end while
```

Figure 33 : Algorithmic sketch for determining Influence

3.2.3 Example: Deriving Influence Using Orkut Data

In this example, we are deriving the influence on George. George has a number of individuals in his social network but the ones, which satisfy the conditions F_1, F_2, F_4 and F_5, are Laura, Peter, Bob, Joe, Kallis and Martin. Though the value F6 (trust) for the 1-hop neighbors is given, we need to derive F6 for the other neighbors. Figure shows F_6 values for all the neighbors whose derivation is shown in appendix C.

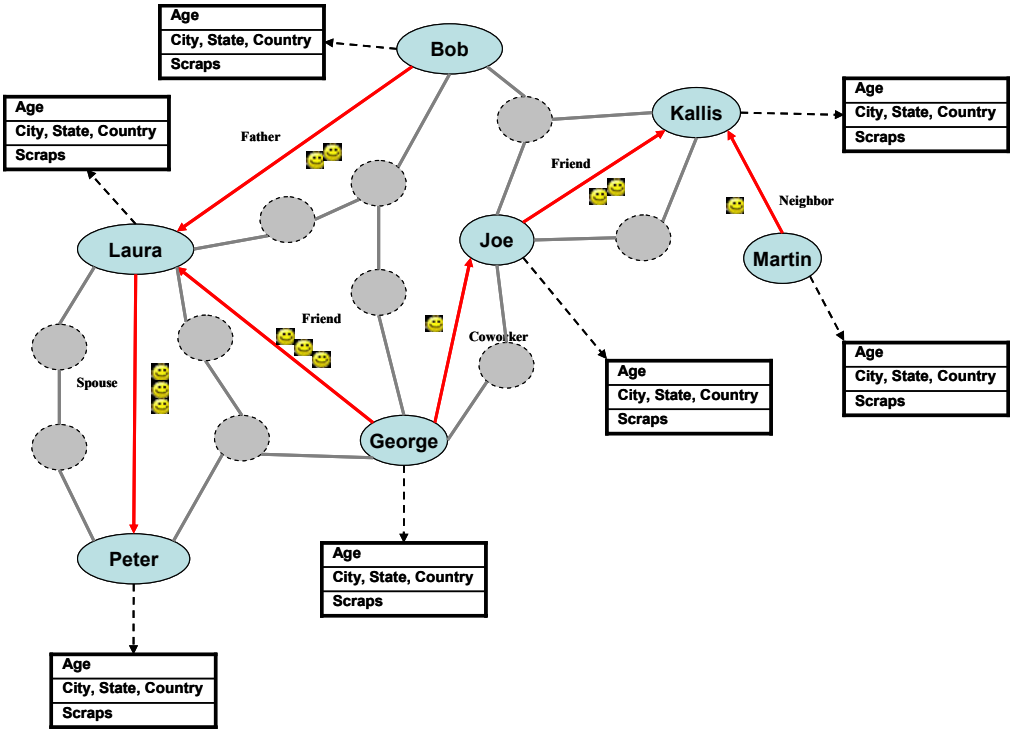


Figure 34 : George's Social Network

Once the F_6 values have been determined, we follow the relationship chains and depending upon the success probability of these chains, the influence each individual has on George is calculated.

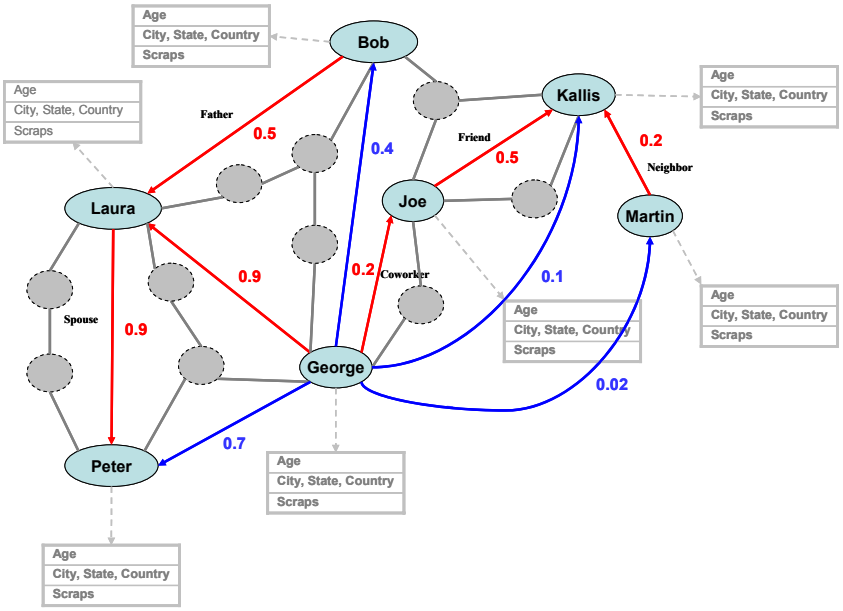


Figure 35 : Deriving trust for more than 1 hop neighbors

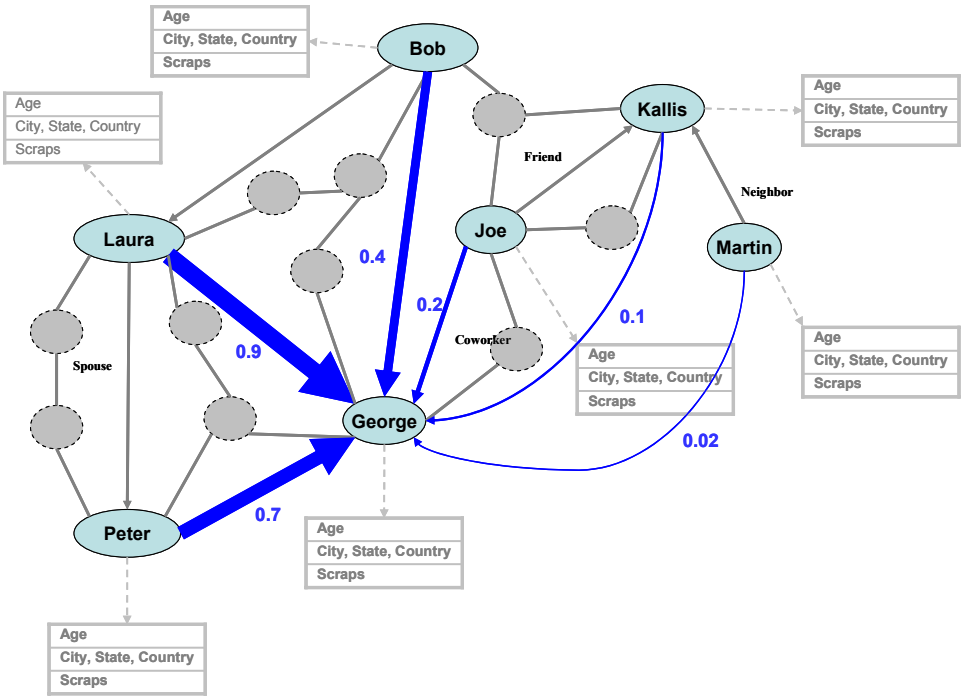


Figure 36 : Influence values

3.2.4 Discussion

There are many potential applications of influence assessment; a few of them could be as follows. If one wants to acquire a government contract, one can derive the influence network of the official. One then knows the people who have the most influence on the official and using them one can indirectly influence the granting of the contract.

The trust network of an organization can be used to promote an agenda/idea in a community. One pinpoints the person in the community who is trusted by most of the members and makes him believe in the agenda/idea that one wants to propagate. Since many people trust him, he implicitly has influence over them. Thus, one has managed to reach a large audience without much effort.

Since the application is based on graph traversal, one can use the depth-first search and breadth-search algorithms, but the rules for traversal would depend upon adherence to the con-

straints specified in the earlier sections. The time complexity in worst case would be O (|v| + |e|) plus the time required for string matching. The number of nodes and edges, which would be required to be examined, is constrained by the length of the relationship chains one proposes to follow. In most of the scenarios, the maximum length will not be more than three. If the size of the graph is not pruned then iterative deepening depth first search can be used but then the time complexity would be $O(b^d)$ where b is the branching factor and d is the dept of the shallowest goal state [14].

3.3 Social Linkage Analysis

Social linkage analysis is aimed at finding the most effective relationship chain between two individuals in a social network. In social linkage analysis based application, the relationship graphs of two individuals are considered. It is a bit more complex than social fabric analysis since here the scope of the relationships under consideration increases two folds. The computation finds the most efficient path between a source and a destination.

In a social network, a person can establish contact with another person by following a number of different relationship chains. However, the objective is to choose the chain, which maximizes the probability of contact and the success of forging a relationship while satisfying certain constraints. The purpose for establishing contact could be varied. Thus, one can see that it is fundamentally a maximum flow problem within a pruned graph. We have illustrated this analysis using an example of a person trying to force himself/herself called vested socialite into someone else's social network. In section 3.3.1 we clearly define a vested socialite and the social linkage analysis, he/she carries out in-order to achieve his/her objective. The following sections 3.3.2

and 3.3.3, present an algorithmic structure and an example demonstrating the application of the algorithmic structure.

3.3.1 Example: Vested Socialite

It is always advantageous to have an influential social network or be part of an influential person's social network. Being an associate of important people can help one in a variety of ways. However, what if one is not part of such a social network but wants to get into one of them? We call such a person who is trying to force himself/herself into someone's social network as a Vested Socialite.

Wikipedia defines a socialite as "a person (male or female, but more often used for a woman) of social prominence who spends a significant amount of his or her time and resources entertaining and being entertained. A socialite is usually a member of the upper class or aristocracy." The vested socialite is different from the classical one. He/She does not spend his/her time and resources on entertainment but his/her major efforts are geared towards achieving some underlining agenda, which has a very specific objective.

Whenever one wishes to be part of someone else's social network, the following are the primary factors one considers. We call them V-factors. (i) Objective: The objective decides the target. For example if the vested socialite was looking for job in the hi-tech sector, it would be not as gainful to insert himself/herself into a pharmacist's social network. (ii) Constraining Conditions: There are certain people the vested socialite would like to avoid in the search because their inclusion might jeopardize his/her social insertion process. (iii) Direct Influence: The person whose social network the vested socialite is trying to gain entry into in order to achieve his/her objective should have a reasonably high influence in the decision making process of the social-

ites' objective. (iv)Derived Influence: The person whose social network the vested socialite is trying to gain entry into, should be part of an influential person's social network. (v) Relationship chain: The vested socialite would like to send the introduction message along a chain of relationship where the probability of acceptance is high. (v) Hate List: The vested socialite would like to avoid entering the social network of a person who hates him/her, since that person will never help him/her in achieving his/her objective.

3.3.2 Algorithmic Sketch For A Certain Social Linkage Analysis Based Computation

Once the objective O has been defined, then search for individuals who could be related to that objective, which results in a number of candidates. The aim is to identify a target from these potential candidates.

$$T(O)=maxInfluen\ ce(T_i) \quad (16)$$
$$where\ \ 1\leq i\leq N$$

The influence value can be calculated using the methodologies presented in section 3.2. Once the target has been determined, discover the various paths in the network to reach target "T" from source "S", viz. P1, P2 … PN. Then discard the paths, which do not satisfy the constraints (C1, C2 … CN) and which contain the nodes that are in the enemy list (E) and insert them into chosenPath table.

$$chosenPath\ {}_S^T(O)=P_i \quad (17)$$
$$if\ \ (P_i)\cap(C_i)=\phi\ and\ \ (P_i)\cap(E)=\phi$$

The paths to be followed are selected based on the relationship chains of the individual. The useful relationships for social linkage are the same as that for social fabric analysis. But, there is an increase in the scope to the relations to be considered in addition to the primary relationships, these are uncle (UN), aunt (AN), niece (NI) and nephew (NE). There is an addition in the coworker relationship as well: colleague (CL).

A chain that consists off only friend relationship is the most useful- F{1, }. The second type of chain starting with Friend goes through the friend's relative network F(FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD|) (F|BO|CL|SU). This chain cannot be used beyond 3 hops because further than that the probability of message delivery is low. The third type goes through the friend's coworker network F(BO|CL|SU) .In this type the extent to which one can reach depends upon the type of the second relationship in the chain.

In the chains, which start with a relative relation, the most useful are the ones that have an alternating relative and friend relationship (FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD)[FSOFSO]. The other useful chains starting with a relative relationship are (FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD)F{1,2} ,(FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD)[F],(FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD),(FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD) (BO|CL|SU)[F]. The chain that has coworker as the second link would at the most help you reach a 3-hop neighbor. (FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD)(BO|CL|SU)[F],(FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD) (BO|CL|SU)(FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD)

The coworker chains are useful only over short distances. Following them, most 2-hop neighbors can be reached. The useful chains are (BO|CL|SU)[F] and (BO|CL|SU)(FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD)

One should avoid relationship chains that have an enemy relationship link EN{1, }. In the case where one is looking for a job, one would also like to avoid the following relationship chain (BO|CL|SU) (F|FA|MO|SP|SO|DA|UN|AN|GF|GM|NI|NE|GS|GD). An interesting relationship chain one would like to avoid or would like to keep as a last resort is the one that involves elderly relatives (GF|GM) (F|FA|MO|SP|SO|DA|NE|GS|GD).

```
FindPath(G, start, t, path)
Array paths;
Concatenate (path, start)
If start == end
      return path
for each node n in G
      if(StringCompare(F3(start, n), Enemy)) == false
            if n not in path
                  newpath=FindPath(G, n, t, path)
                  paths.add(newpath)
            end if
      end if
      return paths[ ];

InsertProbabilty(paths[ ], start, t)
      for each path p in paths
            for each node n in path p
                  rchain(p) = rchain(p) + F3(n, n+1)
            StringMatch(rchain(p), patternDb())
            if true then
            insertprobability = patternDb(relationchain(p)).value();
            return insertprobabilty;
            break;
```

Figure 37 : Algorithmic sketch for insertion into a social network

3.3.3 Example: Vested Socialite Network Insertion for securing employment using LinkedIn data

Let us consider an example where John's objective is to secure a job in Microsoft's R & D department. In order for him to achieve his objective, certain conditions should be satisfied: (i) he needs a reference from a Microsoft employee.(ii) his current boss should not be aware that he is looking for a job (iii) the relationship chain from John to the referee should not have a single enemy relation. (iv) The refereeing person should be at an influential position in the R&D department or should have good relationship with the decision makers within the department. (v)

John should not be in the referee's hate list.(vi) John should find such a relationship chain that if he sends a friend request along that chain , the probability of it getting accepted should be the highest among all the paths available.

John performs a search based on Microsoft R&D, which gives him a list of the following people: Victor Bahl, Manish Agarwal, Feng Zhao, Kenneth Weinberg. Out of these results, the most relevant to John's objective is Victor Bahl who is the Principal Researcher/Manager at Microsoft Research.

John discovers a number of paths from him to Victor in using his social network

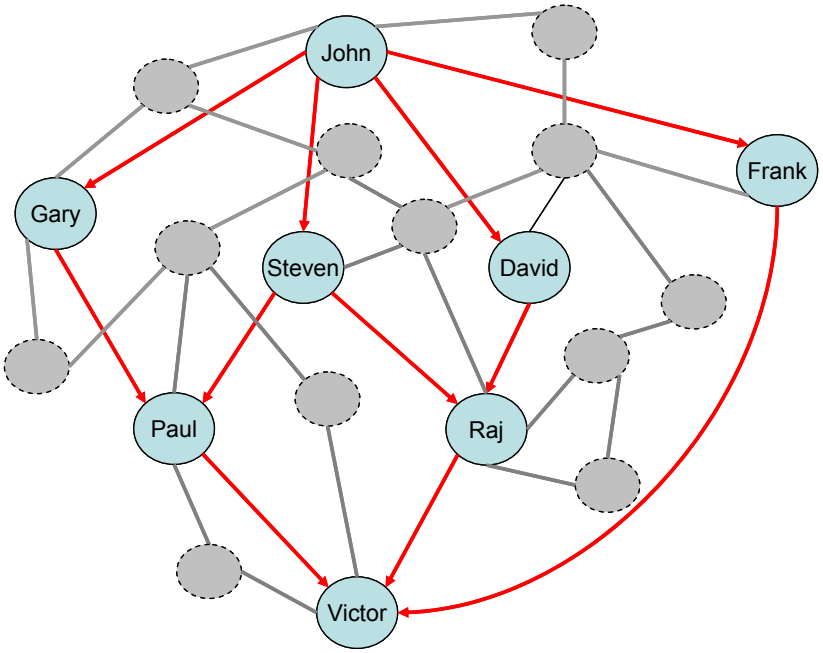


Figure 38 : Intersection of John's and Victor's social network

All of these paths will lead John to Victor but some of them are not feasible. If John sends a friendship request message to Victor along the link, which passes through his enemy, then it is certain that his request would be rejected. Hence, John has to avoid those links. Thus, the next step is to discover the relationship between all the nodes so that John knows which links to avoid.

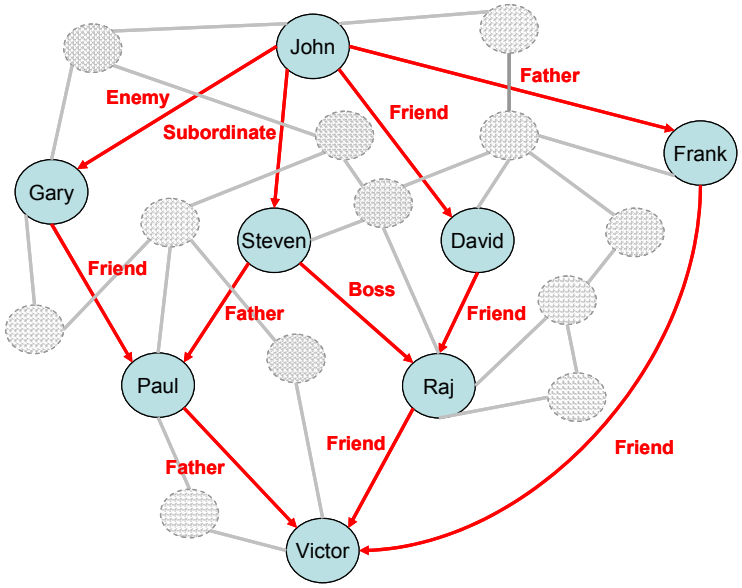


Figure 39 : Relationship structure in John's social network

Once the relationships between all the nodes have been discovered, John realizes that he cannot use the path going through Gary. He would also like to avoid the path through Steven, who is his boss, since he does not want his boss to know that he his looking for a new job. Figure 40 shows the links he should use in red and the ones he should avoid in grey.

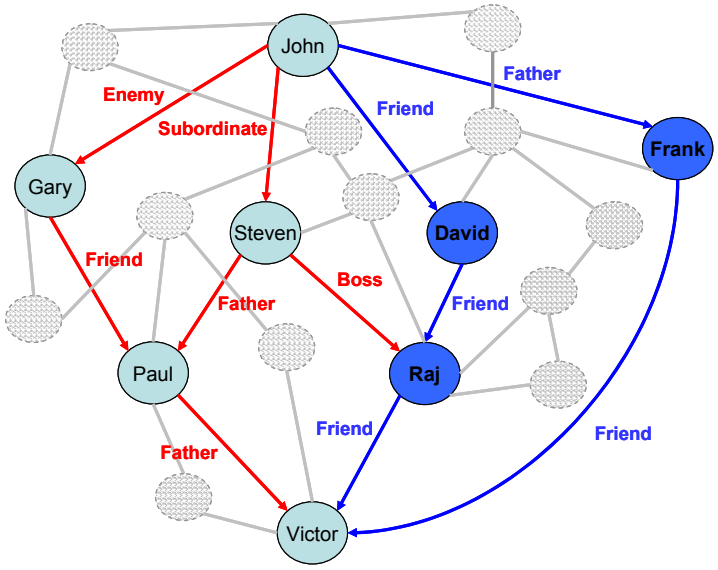


Figure 40 : John's favorable and unfavorable relationship chains

Thus, now John has two possible paths for introduction. He then compares the relationship chains for each path against a database, which gives him a probability of success for each

relationship chain. He chooses the chain with highest probability value and uses it to send the friendship request message. Once Victor accepts John's introduction then the foundation for establishing a relationship between John and Victor has been laid, the only thing left for him to do is to build onto it and finally push through his agenda. Thus, he has achieved his objective of being part of Victor's social network.

3.3.4 Discussion

Lobbyists wanting to get closer to senators can use social network insertion. Companies wanting a sportsperson to endorse their products can use social insertion to get closer to them. The algorithm can be used in applications where we want to select the most favorable relationship chain from among a list of potential chains in order to maximize the probability of success. Since it is a modification of a max-flow problem, a number of algorithms such as brute-force search, Dijkstra's [15] shortest path and Ford-Fulkerson [16] can be used. The complexity of Ford-Fulkerson algorithm depends upon the maximum flow f in the graph and is given as O(E*f). Instead we could use a variation of Ford-Fulkerson, the Edmonds-Karp [17] algorithm which does not depend upon f and runs in O(VE2) time. One can see from our list of constraints that we follow a chain with maximum of three edges; hence, these algorithms can be well suited for analyzing social linkages in a graph pruned using the V-factors.

3.4 Social Ranking Analysis

Social ranking analysis orders individuals in a society based on social properties. The ranking algorithm gives multiple solutions depending upon the different eigen-values. It takes the

complexity of the applications a step further by considering all the individuals in the society and their relationships strength based on various social properties with each other. The social property being considered changes in accordance with the application.

Individuals frequently use some kind of a ranking algorithm while making decisions. Most decision one makes about selection is based upon some kind of ordering. In a community, individuals can be based on various social properties. Depending upon the social property being applied the ranking list changes. An individual who is top most in the influential list may not be at the top in the trustworthy list. Thus, ones order criteria and hence ones list changes depending upon the social property being considered while ranking individuals. In the next section 3.4.1, we present how trust can be used to provide ranking within a community and provide an algorithmic structure for applying eigen-value computation to come up with this ranking. Finally, we support our claims by working out an example in section 3.4.3.

3.4.1 Example: Ranking Based On Trust

In order to find the most trustworthy person in an individual's social network two groups of individuals are considered. The first group is of the individuals who trust him/her and the second group consists of individuals he/she trusts. The first group is considered since the more people trust someone the more trustworthy he is. The reason for considering the second group is not that obvious. If a person trusts trustworthy people then one knows that he is trustworthy but what if he is showing a high degree of trust in non-trustworthy people. This projects an aberration and gives one an indication that something is wrong with him. Thus, we introduce two terms, source rating and sink rating to represent the above phenomenon.

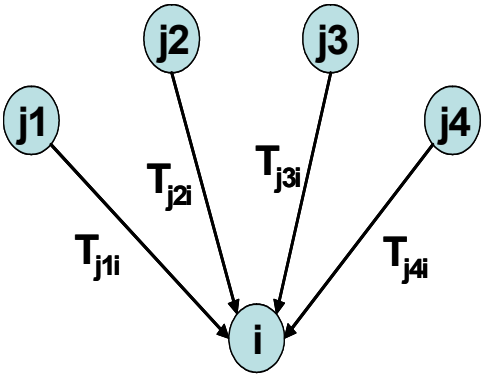


Figure 41 : Sink Ranking

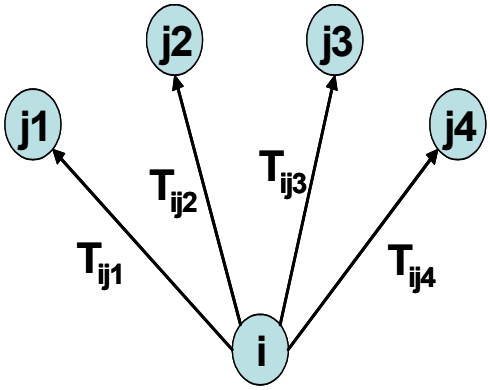


Figure 42 : Source Ranking

3.4.2 Algorithmic Sketch And Methodology

The sink ranking for i is given by equation 18 for some $\lambda > 0$.

$$x_i = \lambda^{-1} \sum_{j=1}^{n} T_{ji}\, x_j \tag{18}$$

In the matrix form it can be represented as in equation 19.

$$Tx = \lambda x \tag{19}$$

Where T is the adjacency matrix of the social network graph, whose elements are T_{ij}, and x is the vector whose elements are x_i. The rankings we are looking for are an eigen-vector of the adjacency matrix with eigen-value λ.

The source ranking for i is given by equation 20 where T^T is the transpose of T.

$$y_i = \mu^{-1} \sum_{j=1}^{n} T_{ij}^{T}\, x_j \tag{20}$$

A generalization for equations 19 and 20 can be given as follows.

$$T^T x = \mu\, y \tag{21}$$

Eliminating y from equation 21 gives us.

$$T\ T^T x = \lambda\ \mu\ x \tag{22}$$

Then λ is an eigen-value of T. Solving the characteristic equation for λ gives the eigen-values of T. Once an eigen-value is determined, it may be substituted into equation 19, and then that equation may be solved for the corresponding eigenvectors. The characteristic polynomial of T is det (A – λI).

3.4.3 Ranking Based On Trust Numerical Example

We consider a small network as in figure with the numerals along the edges denoting the level of trust one node has in the other node then by applying eigen computation we can derive a rank for each node with respect to trust.

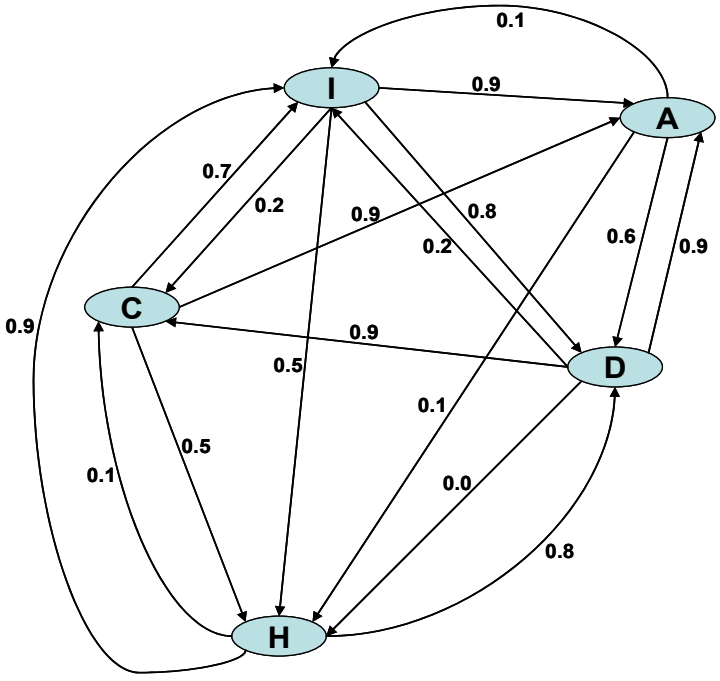


Figure 43 : Complex social network

For different eigen-values λ1, λ2, λ3 greater than zero the eigen vectors X1, X2, X3 are as follows

λ1 = 2.6941, X1 [A, D, C, I, H] = [0.2139, 0.4313, 0.4813, 0.5249,0.5109].

λ2 = 0.3568, X2 [A, D, C, I, H] = [-0.0719, -0.0514, 0.1713, -0.6125,0.7666].

λ3 = 0.9537, X3 [A, D, C, I, H] = [0.4444, 0.0926, -0.4508, 0.0073,-0.7685].

Thus we can see that the eigen computation gives us three possible ranking for the nodes.

3.4.4 Discussion

We can use the method for ranking a community based on any social property such as status, influence, hostility, comical, friendliness. We need a adjacency matrix and the ranked list can be deduced using eigen-value computation. Eigen-value and eigenvector computation are expensive and require O(n¬3) operations.

3.5 Placement Within A Community

Community placement is the problem of finding highly connected individuals within a community who could influence a majority of the community to agree on or believe in an issue.

Propagandists use a naive brute force method for promoting their cause in a community. They contact each individual personally and try to explain and promote their objective. A better approach would be to determine a set of high ranked individuals and use them and their contacts to propagate their agenda. This is an extension of the ranking problem. Here in addition to finding the highly ranked individual, we should also be aware that those individual satisfy certain conditions.

3.5.1 Example: Multi-Faith Group

The aim of the application is to gather a group of individuals who belong to different faiths and use them to propagate religious tolerance among the community. The aim is two-fold , first identifying the individuals and second to ensure that everyone in the community is influenced.

The following are the requirements for including individuals in the multi-faith group. (i) The individuals should be from different faiths (ii) The individuals should be highly connected (iii) The individuals should have friends in multiple faiths.

All of the above information can be gathered from the individual's profile on Orkut. The individual's faith can be found from the religion section. The connection density can be judged by observing the number of friends an individual has. We can determine the faith diversity in ones friend's network by performing a search based on religion on ones friends network. Orkut provides these facilities. Using this information one can generate a graph of a community with nodes representing the individuals and the edges representing the relationships between them. An algorithm is applied on this graph to generate a clique set consisting of individuals who will propagate the agenda.

3.5.2 Algorithmic Sketch

The algorithm consists of two main stages. The first stage is ranking the nodes and the second stage is the propagation of the agenda within the network using the highly ranked nodes as source.

The ranking algorithm is first used to rank every individual in the graph. Then we pick the top 10% of the individuals, they are called the Dominators. Each dominator then sends the

agenda to all its neighbors. A data structure is maintained which stores the node name, the level of influence over it and the influencing dominator. The last two entries are updated if while traversal we find a dominator who is more influential than the earlier one. Thus, using this heuristic algorithm, the propagandists just have to inject their message into these few individuals and they would do the job of spreading it in the rest of the community.

```
Aim : Clique (G, CS_I, CS_R,β)
    Set S = PickNeighbors (G, CS_I);
    Array priority [n] = prioritize (S);
    Array convertDb[n][3];
    for (i=0; i < priority.size ; i++)
        convertDb[i][0] = priority[i];
    end for
    while G is not empty
        m = max( priority [n] )
        R = R U m
        G = G – m
        call sub evaluate (m, CS_R)
        if min(convertDb()) > β
           break;
        end if
    end while
    return R;
    sub evaluate (m, CS_R)
        neighbor [ ] = PickNeighbor (m, G, CS_R)
        for (j = 0 ; j < neighbor.size; j++)
          if (convertDb[neighbor[ j ] ][1] < Influence(m,neighbor[j]))
                convertDb[neighbor[ j ] ][1] = Influence(m,neighbor[j]);
                convertDb[neighbor[ j ] ][2] = m;
          end if
        end for
    end sub
```

Figure 44 :: Algorithmic sketch for placement within a community problem

3.5.3 Numerical Example

Our aim is to select the appropriate individuals from the network shown in figure 45 and ensure that everyone in the community receives the message

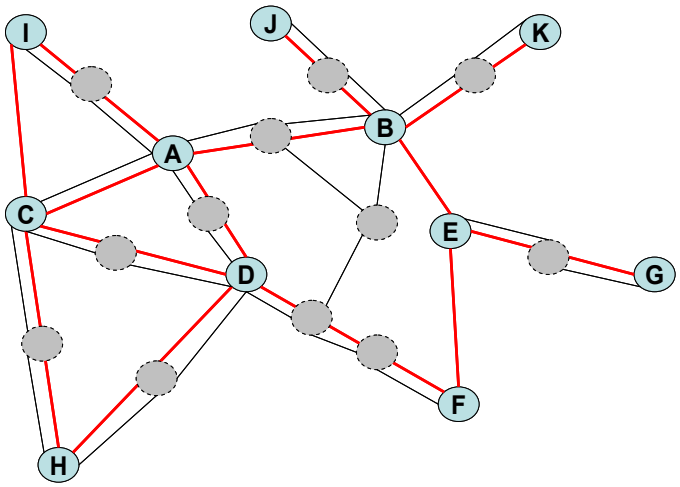
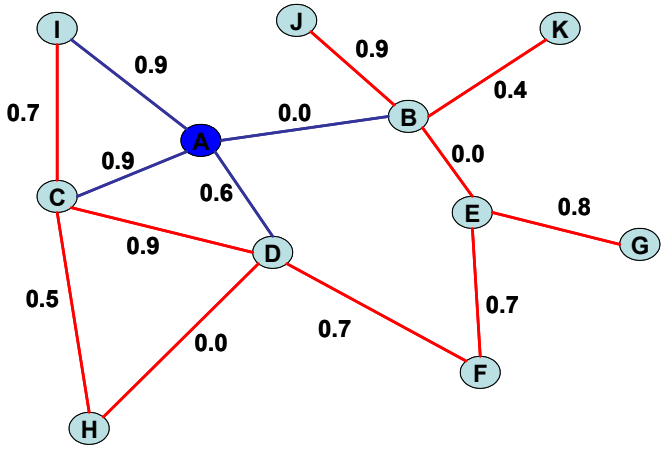


Figure 45 : Sample Social Network

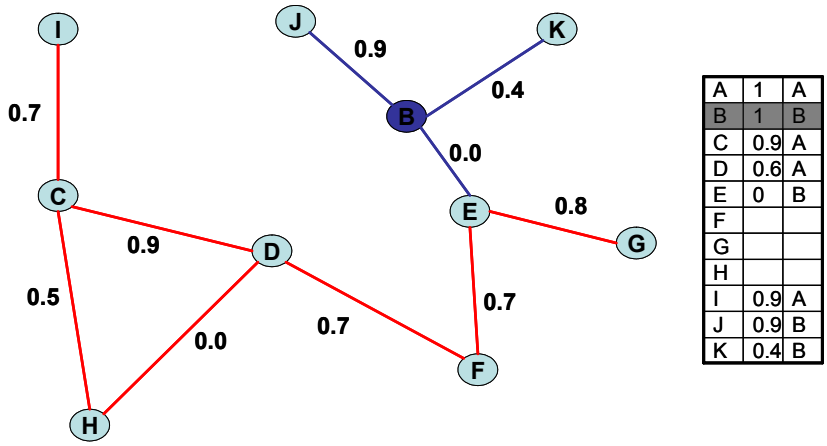
At first, we derive the influence relationship between all the individuals. Individual A is the most connected individual in the network and hence is selected first. We update the data structure with the influence value and the dominator.



A	1	A
B	0	A
C	0.9	A
D	0.6	A
E		
F		
G		
H		
I	0.9	A
J		
K		

Figure 46 : Computation Step 1

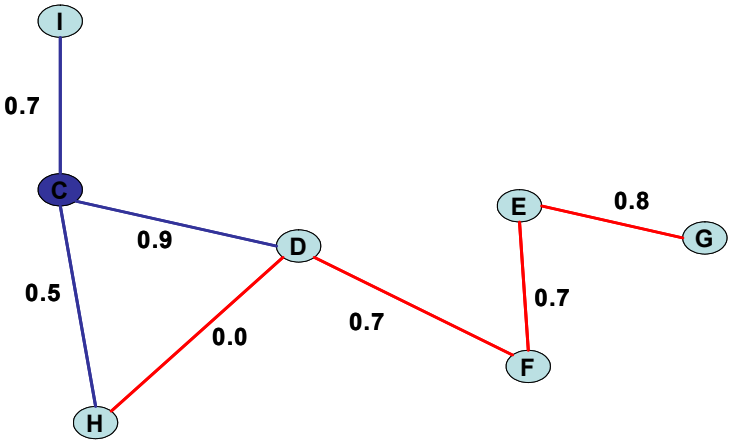
Once the influence of the first dominator has been recorded then it is removed from the graph and then we traverse to the next dominator, in our example it is B.



A	1	A
B	1	B
C	0.9	A
D	0.6	A
E	0	B
F		
G		
H		
I	0.9	A
J	0.9	B
K	0.4	B

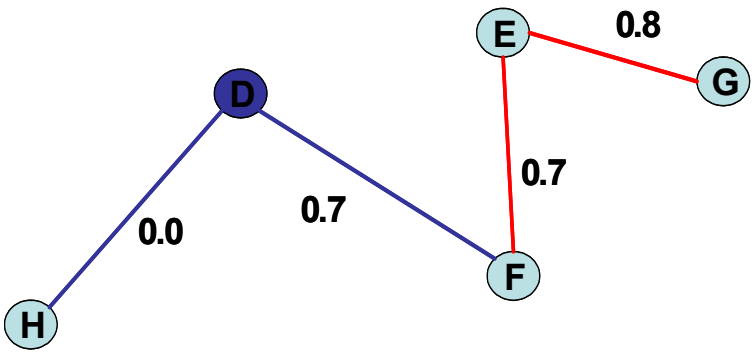
Figure 47: Computation Step 2

This process is repeated recursively until all the values in the influence column are above a certain threshold β. The multi-faith group consists of the dominators in the third column of the data structure.



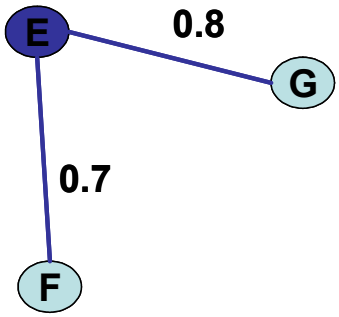
A	1	A
B	1	B
C	1	C
D	0.9	C
E	0	B
F		
G		
H	0.5	C
I	0.9	A
J	0.9	B
K	0.4	B

Figure 48 : Computation Step 3



A	1	A
B	1	B
C	1	C
D	1	D
E	0	B
F	0.7	D
G		
H	0.5	C
I	0.9	A
J	0.9	B
K	0.4	B

Figure 49 : Computation Step 4



A	1	A
B	1	B
C	1	C
D	1	D
E	1	E
F	0.7	D
G	0.8	E
H	0.5	C
I	0.9	A
J	0.9	B
K	0.4	B

Figure 50 : Computation Step 5

3.5.4 Discussion

The algorithm can be used in propagation of health awareness in a community. It can be used in the distribution of aid in case of natural disasters. The algorithm can be used in scenarios where we want faster distribution of anything in a community. The Belief Propagation algorithms can also be used for the propagation of agenda. The complexity of the algorithm would be O(n3) needed for ranking and O(|V|+|E|) needed for the traversal of the graph.

3.6 Game Theory

Game Theory is another computational paradigm, which may eventually find applications in social computing. Game theory has been defined as a mathematical method of decision-making in which a competitive situation is analyzed to determine the optimal course of action for an interested party, often used in political, economic, and military planning. [19]. The games that are studied by game theory have been precisely defined mathematically. A game has a set of players, a set of moves that the players can employ, and a specified payoff for each strategic move. In literature, the games are represented in two main forms, namely normal form and extensive form, and there are five types of games which are as follows. (i) Symmetric Game: A symmetric game is a game where the payoffs for playing a particular strategy depend only on the other strategies employed, not on who is playing them. Common example of symmetric game is the prisoner's dilemma . The best deterministic strategy for solving prisoner's dilemma has been found to be "Tit for Tat" developed by Anatol Rapoport. [20] (ii) Zero Sum: A zero-sum game describes a situation in which a participant's gain or loss is exactly balanced by the losses or gains of the other participant(s). The best example of zero sum game is Poker since in poker an individual wins the exact amount his/her opponent loses. The fundamental theorem of poker proposed by David Sklansky is the starting point for many poker strategies. (iii) Sequential Game: A sequential game is a game where one player chooses his action before the others choose theirs. Combinatorial games such as Go are examples of sequential games. Berlekamp and Wolfe developed effective strategies for playing and wining at Go. (iv) Perfect Information: It is a subset of sequential game where one has perfect information about the game. A game is one of perfect information if all players know the moves previously made by all other players. All sequential games can be examples of perfect information. (v) Infinitely long games: The games which last

for many moves and the winner is not known al those moves have been completed fall under this category.

Game theory has been applied in social sciences, and is now finding use in other academic fields as well [20]. It has become a topic of interest for computer scientists because of its application in artificial intelligence and cybernetics. It provides a means to formulate, analyze and understand strategic scenarios. Although it is a computation paradigm but it differs in a fundamental way from the previous five computations identified in the previous sections. It focuses on strategy analysis of the intentions and actions of the participants while the previous computations focus on structural analysis of the social network. For that reason, the scope of this thesis does not focus on game theory although it can be combined with structural analysis in the future. In the following section, we explain how game theory can be used in a few of the social network based computations introduced in the preceding sections.

The basic requirements for use of game theory are that there should be a game and there should be at-least two individuals playing that game and their actions should be interdependent. Due to these constraints, game theory does not compliment with social profile mining, social fabric analysis and social ranking analysis but it can be used in social linkage analysis, and placement within the community.

Game theory may not be profitable applied to social profile mining because the computation essentially does data mining, which is purely statistical analysis. It probably cannot be used in social fabric analysis either since those classes of computation deal with the neighborhood of a single individual and his/her primary relationship interactions. Social ranking analysis does an ordering of individuals hence, there is no interactive game played due to which game theory here can be applied. Opinion used in ranking can be refined using game theory with due consideration to grouping.

Social linkage analysis could use backward induction [21] to determine the strategy that would output the most effective social path between the two individuals, so that the payoff for the involved individuals is maximized. Placement within the community can use game theory to maximize the effect of influential individuals by strategically selecting them in such a way that a few can persuade many. The game that is being played in this case is maximizing a global social property. The individuals involved in the game belong to two groups. One group consists of a few influential individuals and the other group consists of the target community. The strategy should be to maximize the influential circle of each member of the first group and at the same time ensure that almost the whole community is influenced.

In the scenarios where a recommendation function uses the opinions of individuals to evaluate the rating of a particular person can be considered as a very elementary application of game theory. One such function is the Generic Reputation Function discussed in sections 2.3 and 2.4. This function may be further refined to incorporate complex game theoretic concepts.

3.7 Conclusion

Over the last couple of years, a number of social networking websites have become the cornerstones for social interactions. The information being shared by individuals on these websites is becoming richer by the day. In this paper, we have presented a variety of applications in which the information provided on the social networking websites can be used in a variety of diverse applications. Here we give a broad classification of the applications that can be conceived by monitoring and using the social interaction between individuals on social networking web sites. Humans use some kind of high level algorithm while interacting with fellow beings. Even though it is difficult in replicating the exact human social interaction algorithms, here we present

a few algorithms, which are machine understandable and replicate to some extent the social interactions and relationship physics, which goes into deriving social properties such as influence, trust, reputation and status.

CHAPTER 4

EXAMPLES OF SOCIAL NETWORKS

4.1 Language Graph Of A Publication Network

The major entities involved in the publication network are the Authors, the Organizations, the Paper, the Journal, the Reviewers, the Editors and the Topic Area. The topic area is the focal point of the network. It is evident from the language graph presented in figure 51 that there exist direct relationships between various nodes of the graph. These are visible relationships, our aim is to find the ones, which are not obvious but have a profound affect on the publication decisions. For example, the relationship between the author of a paper and the editorial board of the journal in which the paper has been submitted for publication. We expose these relationships using the example graph of a publication network shown in Figure 52. We would be using the instance graph for illustrating how we can use the relationship algebra presented in chapter 2 for useful purposes such as "Finding a set of reviewers for a particular paper". The only relationships available to us are the ones indicated by the arrows going from the source to the sink. For example, the arrow drawn from "Jeroen Dietz" to "P_1" represents an Author → Paper relationship. The author → organization relationship can have many flavors such as student, professor, boss, employee, and researcher. This is because the organizations can be varied, such as universities, research labs, private companies, government funded establishments etc. Using the primary relationships enumerated in the table 15 and applying the relationship algebra upon them enables us in deriving and detecting interesting phenomenon such as conflict of interest.

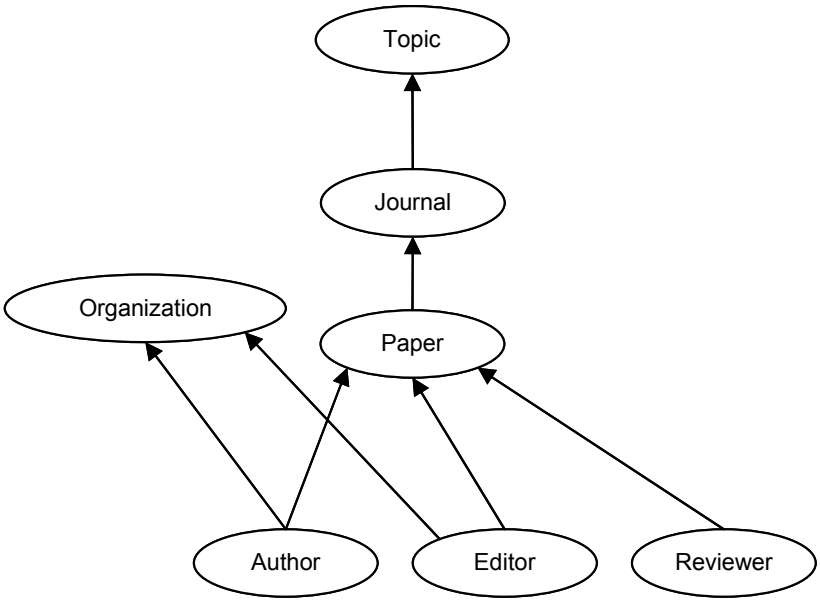


Figure 51 : Language graph of publication network

4.1.1 Application: Reviewer Selection

The network can be used to select a set of authors who can be on the reviewing committee of a paper such that they meet certain restrictions. The reviewer selection can be expressed by a set of constraints. Below is an example set:

Reviewer Selection Constraints: (i)The reviewer should not be a coauthor of the paper he is going to review, (ii) he should not be a coworker of the author for example the author and the reviewer should not be faculties in the same university. (iii) The reviewer should not have submitted a paper in the same journal or conference and (iv) finally he should be well acquainted with the subject area being discussed in the paper.

Primary Relationship	Notation
Journal → Topic Area	$M_{J_i}^{J-T}$
Editor → Paper	$M_{E_i}^{E-P}$
Paper → Journal	$M_{P_i}^{P-J}$

Author → Paper	$M_{A_i}^{A-P}$
Reviewer → Paper	$M_{R_i}^{R-P}$
Author → Organization	$M_{A_i}^{A-O}$

Table 14 : Primary Relationships for a Publication Network

We have illustrated the complete reviewer selection process by working out an example in appendix A using the network shown in figure 52

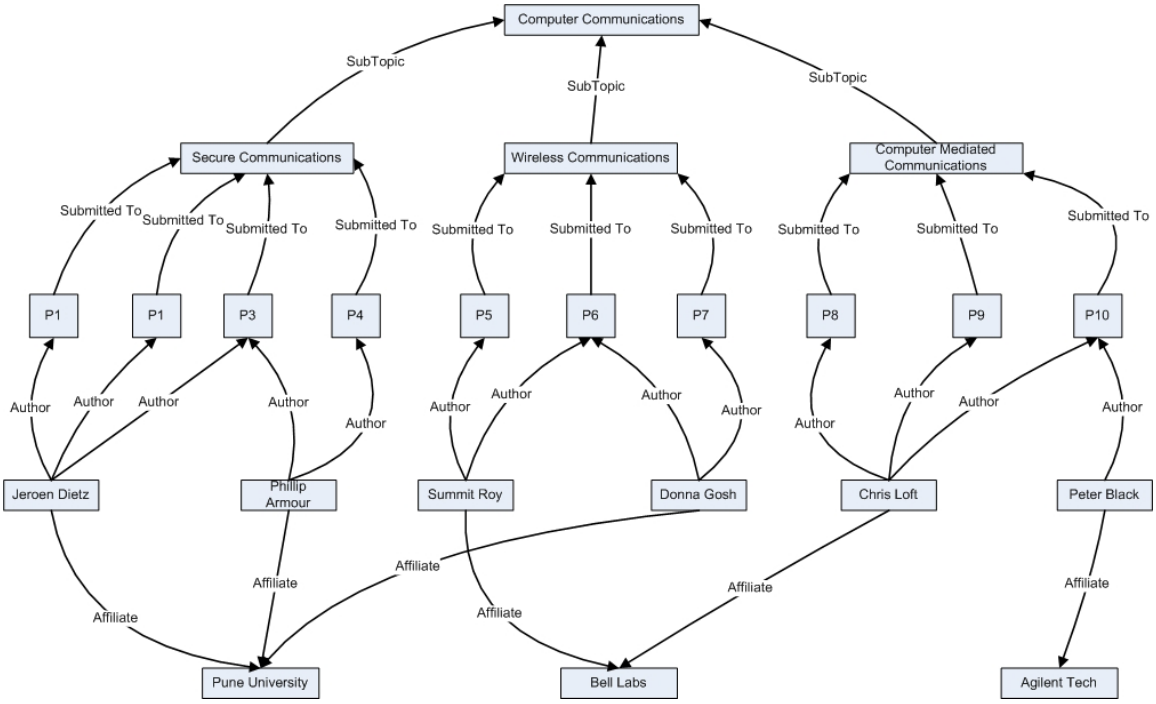


Figure 52 : Instance graph for the publication network

The aim of the example is to find a reviewer set for papers P5 and P6 from among the four authors available. The first step is to determine the authors and coauthors for P5 and P6 . This is achieved by multiplying the matrix M_A^{A-P} ,which represents the relationships between the authors and the papers with its transpose matrix. The resultant matrix $M_{coAuthor}$ represents the co-author relationship between the respective authors. In the next step we determine, which authors have submitted papers in the same journal. In order to determine this we first need to establish a relationship matrix depicting the relationship between the authors and the journals M_A^{A-J} ,

which is done by multiplying the matrices, M_A^{A-P} and the one representing papers-journals relationships M_P^{P-J} . The resultant co-journal matrix $M_{coJournal}$ is a product of M_A^{A-J} and its transpose matrix. Now we have to determine, which all authors are coworkers. The coworker matrix $M_{coWorker}$ is computed by multiplying the matrix representing the authors-organization relationships M_A^{A-Org} and its transpose. Finally we determine the non conflict of interest matrix $M_{nonConflict}$ by subtracting each of the *coAuthor*, *coJournal* and *coWorker* matrices from the matrix depicting the relationship between all the authors belonging to the same topic area Mall. The reviewer set matrix $M_{reviwer}$ is calculated by multiplying the $M_{nonConflict}$ and the matrices $\left(M_A^{A-P}\right)^T$. Applying the row extraction set operation ρ on the reviewer set matrix gives us the reviewer set for papers P_5 and P_6.

$$M_{reviewer} = M_A^{A-P}\left[M_{all}\theta M_{coAuthor}\theta M_{coJournal}\theta M_{coWorker}\right] \tag{23}$$

$$ReviewerSet(P_i) = \rho_i^{M_{ij}=1}\left(M_{reviewer}\right) \tag{24}$$

In the above discussion we have mentioned the term conflict of interest , which can be defined as follows. A conflict of interest consists of three entities, the source "i", the sink "j" and the relationship between them "R". It occurs if we have two distinct relationship trails R_1 and R_2 from i to j and their intersection set is nonempty.

$$S_i^j(R_1) \cap S_i^j(R_2) = \phi \tag{25}$$

We can determine the conflictset for each author in the above example by applying the column extraction set operation Ψ on the reviewer set matrix $M_{nonConflict.}$

4.1.2 Application: Panel Selection

Another use of the network can be in the selection of intellectuals for the formation of a panel for a particular research area. The panel should satisfy the following constraints. For our example, the constraint set is given below.

Panel Selection Constraints:(i)The members of the panel should consist of people from different fields of the area and (ii)they should belong to varied organizations such as universities, research labs, industry etc.

The algorithm for panel selection is as follows. The first step is to extract the expert in each field from the M_A^{A-J} matrix using the max column set operation "ξ" to form the *expertset*. Then for each panelist in the extracted set we determine the kind of organization represented by him or her through the M_A^{A-Org} matrix. The zero column set operation is applied to the M_A^{A-Org} matrix to ascertain that all organizations have been represented on the panel. If the operation results in a nonempty set, then a person from the missing organization is picked from the M_A^{A-Org} matrix using row extraction to give the *missingexpert* set. The panel is the union of *expertset* and *missingexpert* sets.

$$expertset\ (T) = \xi_j(M_A^{A-J}) \tag{26}$$

$$missingOrg\ (T) = \theta(M_A^{A-Org}) \tag{27}$$

$$missingexpert(T) = \rho_{missingorg}^{M_{ij}>0}(M_{Ai}^{A-Org}) \tag{28}$$

$$panelset(T) = expertset(T) \cup missingexpert(T) \tag{29}$$

4.2 Language Graph Of A Social Network

An individual's social network primarily consists of family, friends, neighbors, coworkers and the organizations with which he is affiliated. The circles denoted by A, B, C and D are the individuals in a community. The ellipses denote the six major types of relationships we have considered for our example. The smaller rectangles within the ellipses denote the refined relationships for each class. These refined relationships are enumerated in a table below.

Relationship Class	**Relationship Objects**
Family	Father, Son, Daughter, Spouse
Friend	Good Friend, Acquaintance
Enemy	Competitor, Contradicting Beliefs
Coworker	Boss, Colleague, Subordinate, Partner
Neighbor	Next Door, Same Community
Organization	Educational, Religious, Entertainment, Philosophical

Table 15 : Refined Social Relationships

The instance graph shows the social network of an individual "George". George's social network consists of his family, his fellow workers, his neighbors, the organizations he is associated with and his enemies. Each of these nodes further has their own social networks, which are a part of George's network, but George has a derived relationship with the nodes of these secondary networks. The strength of George's derived relationships depends upon the strength of his primary relationships.

Primary Relationship	**Notation**
Individual → Company (Owner)	$M_{Ind}^{Ind-Company}$
Individual → Friend	$M_{Ind}^{Ind-Ind(Friend)}$
Individual → Father	$M_{Ind}^{Ind-Ind(Father)}$
Individual → Org (Member)	$M_{Ind}^{Ind-Org}$
Individual → Neighbor	$M_{Ind}^{Ind-Ind(Neighbor)}$
Individual → Enemy	$M_{Ind}^{Ind-Ind(Enemy)}$
Individual → Boss	$M_{Ind}^{Ind-Ind(Boss)}$
Individual → Coworker	$M_{Ind}^{Ind-Ind(Coworker)}$

Individual → Spouse	$M_{Ind}^{Ind-Ind(Spouse)}$
Individual → Org (Client)	$M_{Ind}^{Ind-Org(Client)}$

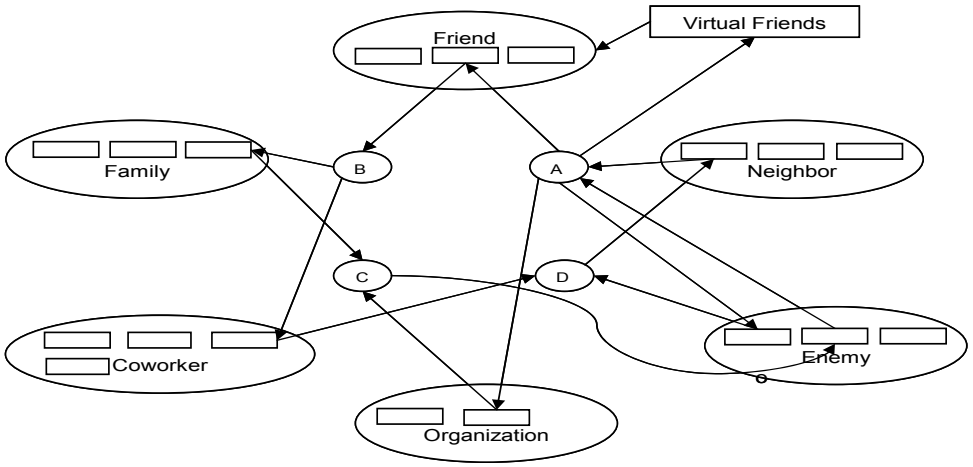


Figure 53 : Language Graph of a Social Network

4.2.1 Application: Immunization

Suppose George is a virus carrier and we want to find out the people who might have been infected by him and need vaccination. In order to achieve this we need to determine the vaccination set from George's social network. The people to be included in the set should satisfy certain conditions. For our particular example, the conditions are as follows:

Immunization Constraints: (i) The people most vulnerable are the ones which come in physical contact with George. These are usually friends, family, neighbors and coworkers. They have to be immunized.(ii) The second group of people who are likely to get infected are the ones which belong to George's derived network i.e. his greater than 1 hop neighbors. The likelihood of them been infected depends upon their relationship strength with George's 1 hop neighbors. For our example, the threshold value is 0.6

Using these two relationship strengths, we compute the matrices $M^{Spouse}_{George}, M^{Father}_{George}, M^{Neighbor}_{George}, M^{Coworker}_{George}, M^{Friend}_{George}$, which each represents the derived relationship strength between George and his greater than one hop neighbors. Then we apply column extraction for the first row of each of these matrices to get an individual subset.

$$set\left(A\right)^{Spouse} = \psi^{M_{ij} > 0.6}_{A}\left(M^{Spouse}_{A}\right) \tag{30}$$

The final vaccination set is a union of all the individual subsets.

$$vaccinationset(A) = \{set(A)^{Spouse} \cup set(A)^{Father} \cup set(A)^{Friend} \cup set(A)^{Coworker} \cup set(A)^{Neighbor} \cup set(A)^{Son}\} \tag{31}$$

4.2.2 Application: Crime Watch

The network can be used for crime prevention. Using the network information the police can get together a surveillance team that would help to keep a watch on the places likely to be visited by the fugitive. The fugitive will almost certainly receive help from his family members and friends. The constraints to become a member of our surveillance team are as follows.

Surveillance Team Constraints: (i) The team member should a neighbor of either fugitive's family members or his friends. (ii) He should be a friend of the fugitive's family or fugitive's friends.

The surveillance set can be found by determining the relationship matrices between the fugitive say "George" and the neighbors of his family and friends. The matrices $M^{Neighbors(Friends)}_{George}$ $M^{Neighbors(Family)}_{George}$ represent the relationship between George and the neighborhood of his family and friends. The surveillance matrix $M^{Surveillance}_{George}$ is the union of these two matrices and we exclude the neighbors who are the fugitive's friends, which are given by the matrices

$M_{George}^{Friend(Family)}$ and $M_{George}^{Friend(Friend)}$ .The surveillance set is obtained from the matrix using column extraction.

$$M_{George}^{Survillance} = \left(M_{George}^{Neighbor(Family)} \oplus M_{George}^{Neighbor(Friends)} \right) \theta M_{George}^{Friend(Family)} \theta M_{George}^{Friend(Friend)} \quad (32)$$

$$survillance set(George) = \psi_{George}^{M_{ij}=1}(M_{George}^{Surveillance}) \quad (33)$$

The social network of George can be used to determine which people and organizations have influence on him. This information is very important if one wants to manipulate his decision on certain matter such that it benefits once interest. The influential set would contain entities, which have a strong relationship with George such as friends, family, church and the ones, which could affect his finances such as business partners, boss, and banks. The first set of individuals can be easily determined since they are direct relationships personal(George). The second set business (A) is determined from the matrix $M_{George}^{Partners}$, which represents his business relationships.

$$influences et(A) = personal(A) \cup business(A) \quad (34)$$

4.2.2 Application: Trust Propagation

As human dependence on the internet as a source of reference before making an important decision increases, there is growing need to differentiate between trustful and distrustful sources. Most of the time we cannot determine it by ourselves, but have to infer it from the experiences of people in our social network. Researchers [22] have studied how to infer trust in a complex relationship network.

There are various forms of trust relationships. The relationship algebra can be used to define various forms of trusts and also determine various combinations and synthesis in a program-

mable way. There are various ways is which trust propagation can be achieved and each individual has a choice as to which path he would take to determine trust. This is because everybody has a different notion of trust and may not share the same principles as someone else does in determining trust.

An example of various trust relations are shown in the social network instance graph of figure 54.

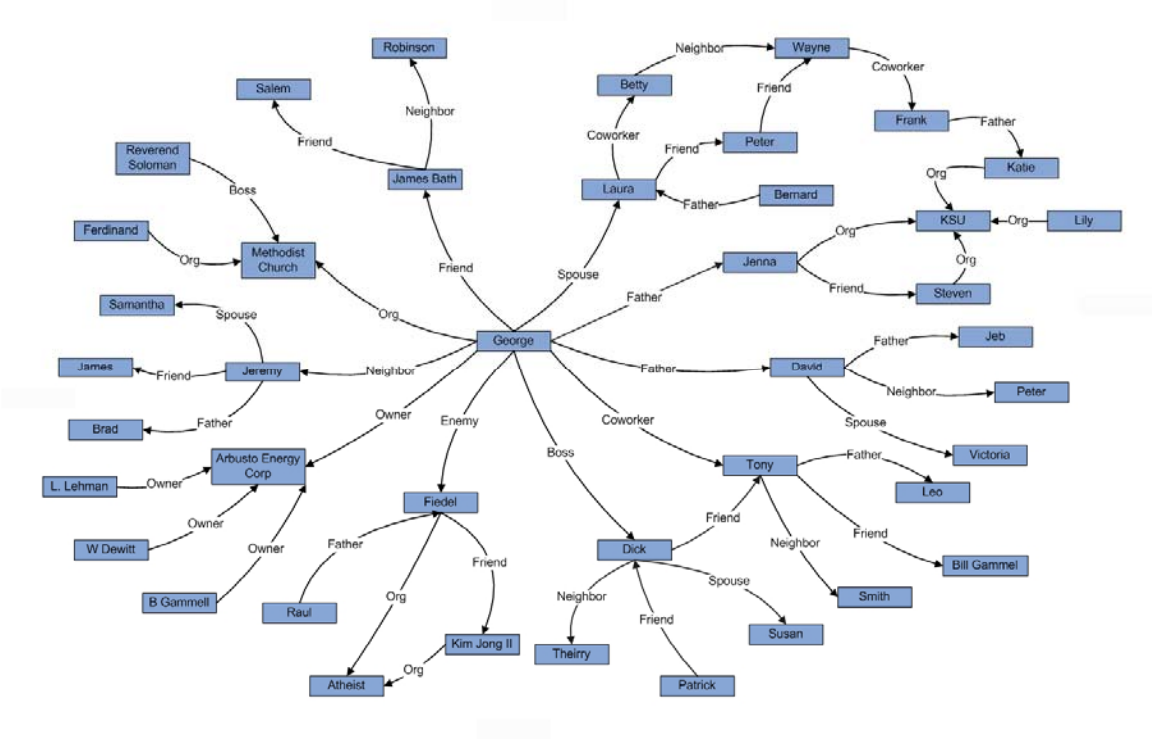


Figure 54: Instance graph of a Social Network

In Appendix B we have worked out an example for illustrating a few of the trust propagation techniques. If George trusts Laura represented by matrix M_{spouse} and if Laura trusts Peter is represented by matrix M_{Friend} then the product of these two matrices M_{result} shows that George trusts Peter. This is an example of *transitive propagation*. If George trusts Laura, Jenna is represented by matrix M_A and Bernard trusts Laura is represented by matrix M_B then the product of M_A , its transpose and M_B shows that Bernard trusts Jenna. This is an example of *inferential*

propagation. If George trusts Laura is represented by M_{spouse} then a product of M_{spouse} and its transpose shows that Laura also trusts George. This is an example of *reflexive propagation*. If George trusts Jenna , is represented by M_{father} matrix and Jenna trusts Kent State University and Lily trusts Kent State University are represented by M_{org} then the product of the three matrices M_{father} , M_{org} and transpose of M_{org} shows that George trusts Lily. This is an example of *trust union propagation*. A complete workout of this example is illustrated in Appendix. B

Atomic Propagation	Result Matrix
George trusts Peter - *transitive propagation*	$M_{result} = M_{Spouse} \times M_{Friend}$
Bernard trust Jenna - *inferential propagation*	$M_{result} = M_{Father} \times M^{T}_{Spouse} \times M_{Father}$
Laura trusts George - *reflexive propagation*	$M_{result} = M_{Spouse} \times M^{T}_{Spouse}$
George trusts Lily - *trust union propagation*	$M_{result} = M_{Father} \times M_{Org} \times M^{T}_{Org}$

Table 16 : Trust Propagation

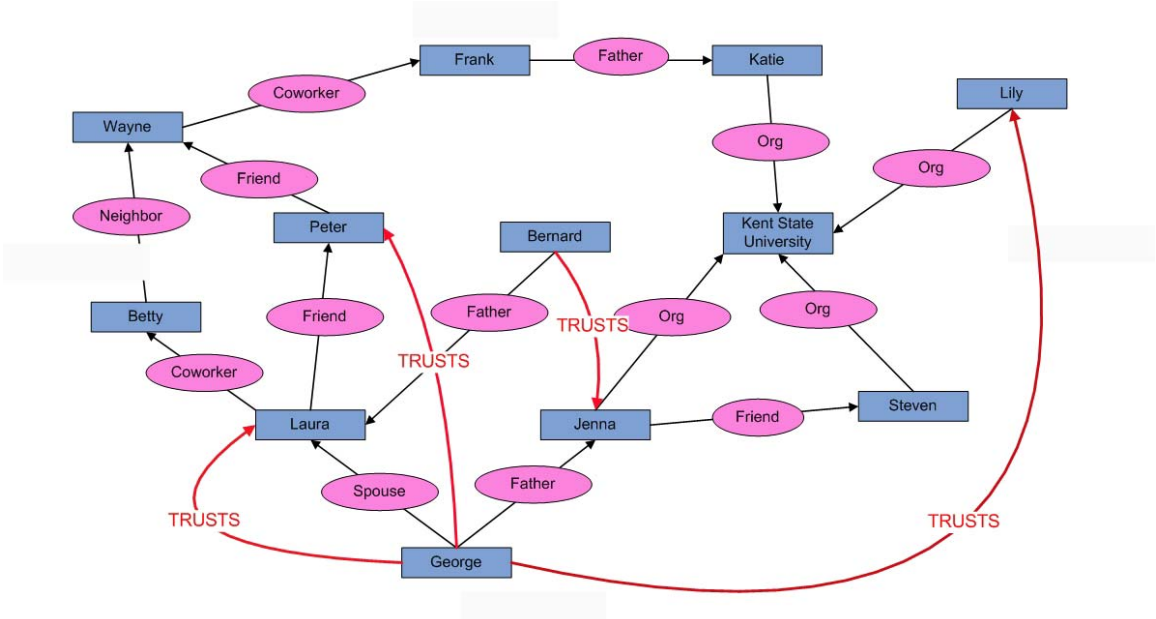


Figure 55 : Instance graph used to demonstrate trust propagation

4.3 Appendix A

Finding A Reviewer Set for papers P_5 and P_6

Step 1 :Derive The Co-Author Matrix $M_{coAuhtor} = M_A^{A-P} \times \left(M_A^{A-P}\right)^T$

	P5	P6
JD	0	0
PA	0	0
SR	1	1
DG	0	1

M_A^{A-P}

X

	JD	PA	SR	DG
P5	0	0	1	0
P6	0	0	1	1

$\left(M_A^{A-P}\right)^T$

=

	JD	PA	SR	DG
JD	0	0	0	0
PA	0	0	0	0
SR	0	0	2	1
DG	0	0	1	2

$\lceil A \rceil^1$ →

	JD	PA	SR	DG
JD	0	0	0	0
PA	0	0	0	0
SR	0	0	1	1
DG	0	0	1	1

$M_{co.Auhtor}$

Step 2 :Derive The Co-Journal Matrix $M_{coJournal} = M_A^{A-J} \times \left(M_A^{A-J}\right)^T$ *Where* $M_A^{A-J} = M_A^{A-P} \times M_P^{P-J}$

	P5	P6
JD	0	0
PA	0	0
SR	1	1
DG	0	1

M_A^{A-P}

X

	S C	W C
P5	0	1
P6	0	1

M_P^{P-J}

=

	S C	W C
JD	0	0
PA	0	0
SR	0	2
DG	0	1

$\lceil A \rceil^1$ →

	S C	W C
JD	0	0
PA	0	0
SR	0	1
DG	0	1

M_A^{A-J}

X

	JD	PA	SR	DG
S C	1	1	0	0
W C	0	0	1	1

$\left(M_A^{A-J}\right)^T$

=

	JD	PA	SR	DG
JD	0	0	0	0
PA	0	0	0	0
SR	0	0	1	1
DG	0	0	1	1

$M_{coJournal}$

Step 3 :Derive The Co-worker Matrix $M_{coWorker} = M_A^{A-Org} \times \left(M_A^{A-Org}\right)^T$

	Pune University	Bell Labs
JD	1	0
PA	1	0
SR	0	1
DG	1	0

M_A^{A-Org}

X

	JD	PA	SR	DG
Pune University	1	1	0	1
Bell Labs	0	0	1	0

$\left(M_A^{A-Org}\right)^T$

=

	JD	PA	SR	DG
JD	1	1	0	1
PA	1	1	0	1
SR	0	0	1	0
DG	1	1	0	1

$M_{coWorker}$

Step 4 :Determine the non-conflict matrix $M_{nonconflict} = \left[M_{all} \theta M_{coAuthor} \theta M_{coJournal} \theta M_{coWorker}\right]$

	JD	PA	SR	DG
JD	1	1	1	1
PA	1	1	1	1
SR	1	1	1	1
DG	1	1	1	1

M_{all}

−

	JD	PA	SR	DG
JD	0	0	0	0
PA	0	0	0	0
SR	0	0	1	1
DG	0	0	1	1

$M_{coAuthor}$

−

	JD	PA	SR	DG
JD	0	0	0	0
PA	0	0	0	0
SR	0	0	1	1
DG	0	0	1	1

$M_{coJournal}$

−

	JD	PA	SR	DG
JD	1	1	0	1
PA	1	1	0	1
SR	0	0	1	0
DG	1	1	0	1

$M_{coWorker}$

=

	JD	PA	SR	DG
JD	0	0	1	0
PA	0	0	1	0
SR	1	1	-2	-1
DG	0	0	-1	-2

$\lceil A \rceil^1$ →

	JD	PA	SR	DG
JD	0	0	1	0
PA	0	0	1	0
SR	1	1	0	0
DG	0	0	0	0

$M_{nonconflict}$

Step 5 :Determine the reviewer matrix for P_5 and P_6 $M_{reviewer} = \left(M_A^{A-P}\right)^T \times M_{nonconflict}$

	JD	PA	SR	DG
P5	0	0	1	0
P6	0	0	1	1

$\left(M_A^{A-P}\right)^T$

X

	JD	PA	SR	DG
JD	0	0	1	0
PA	0	0	1	0
SR	1	1	0	0
DG	0	0	0	0

$M_{nonconflict}$

=

	JD	PA	SR	DG
P5	1	1	0	0
P6	1	1	0	0

$M_{reviewer}$

4.4 Appendix B

Trust Propagation

1.Transitive Propagation $M_{result} = M_{Spouse} \times M_{Friend}$

George trusts Laura. Laura trusts Peter. So George trusts Peter

	George	Laura	Peter
George	1	1	0
Laura	0	1	1
Peter	0	0	1

M_{spouse} X

	George	Laura	Peter
George	1	1	0
Laura	0	1	1
Peter	0	0	1

M_{friend} =

	George	Laura	Peter
George	1	2	1
Laura	0	1	2
Peter	0	0	1

M_{result}

2. Inferential Propagation $M_{result} = M_A \times M_A^T \times M_B$

George trusts *Laura, Jenna*. *Bernard* trusts *Laura* . *Hence, Bernard* trusts *Jenna*

	George	Laura	Jenna	Bernard
George	1	1	1	0
Bernard	0	1	0	1

M_A X

	George	Laura
George	1	0
Laura	1	1
Jenna	1	0
Bernard	0	1

M_A^T X

	George	Laura	Jenna	Bernard
George	1	1	1	0
Bernard	0	1	0	1

M_B =

	George	Laura	Jenna	Bernard
George	3	4	3	1
Bernard	1	3	1	2

M_{result}

3. Reflexive Propagation $M_{result} = M_{Spouse} \times M_{Spouse}^T$

George trusts *Laura.* Hence *Laura trusts George*

	George	Laura
George	1	1
Laura	0	1

M_{spouse} X

	George	Laura
George	1	0
Laura	1	1

M_{spouse}^T =

	George	Laura
George	2	1
Laura	1	1

M_{result}

4. Trust Union Propagation $M_{result} = M_{Father} \times M_{Org} \times M_{Org}^T$

George trusts *Jenna* , *Jenna* trusts *Kent State University* .*Lily* trusts *Kent State University* Hence, *George* trusts *Lily*

	George	Jenna	Lily
George	1	1	0
Jenna	0	1	0
Lily	0	0	1

M_{father} X

	George	Jenna	Lily	Kent State University
George	1	1	0	0
Jenna	0	1	0	1
Lily	0	0	1	1

M_{org} X

	George	Jenna	Lily
George	1	0	0
Jenna	1	1	0
Lily	0	0	1
Kent State University	0	1	1

M_{org}^T =

	George	Jenna	Lily
George	3	3	1
Jenna	1	2	1
Lily	0	1	2

M_{result}

4.5 Appendix C

$$M_{result} = M_{Friend} \times M_{Spouse}$$

	G	L	P
G	1	0.9	0
L	0	1	0.8
P	0	0	1

M_{friend} X

	G	L	P
G	1	0.9	0
L	0	1	0.8
P	0	0	1

M_{spouse} =

	G	L	P
G	1	1.8	0.72
L	0	1	1.6
P	0	0	1

M_{result}

$$M_{result} = M_A \times M_A^T \times M_B$$

	G	L	J	B
G	1	0.9	0.2	0
B	0	0.5	0	1

M_A X

	G	L
G	1	0
L	0.9	0.5
J	0.2	0
B	0	1

M_A^T X

	G	L	J	B
G	1	0.9	0.2	0
B	0	0.5	0	1

M_B =

	G	L	J	B
G	1.85	1.89	0.37	0.45
B	0.45	1.03	0.09	1.25

M_{result}

$$M_{result} = M_{coworker} \times M_{friend} \times M_{friend}^T$$

	G	J	M
G	1	0.2	0
J	0	1	0
L	0	0	1

$M_{coworker}$ X

	G	J	M	K
G	1	0.2	0	0
J	0	1	0	0.5
M	0	0	1	0.2

M_{Friend} X

	G	J	M
G	1	0	0
J	0.2	1	0
M	0	0	1
K	0	0.5	0.2

M_{friend}^T =

	G	J	M
G	1.08	0.45	0.02
J	0.2	1.25	0.1
M	0	0.1	1.04

M_{result}

BIBLOGRAPHY

[1] Gambetta, D., (1988) "Trust making and breaking cooperative relations". New York: Blackwell.

[2] McKnight, D., & Chervany, N., (1996) "The Meaning of Trust". University of Minnesota MIS Research Center Working Paper Series, WP 96-104.

[3] McShane, S., (1995). Canadian Organization Behavior. Ottawa:Irwin.

[4] Meneses, J., (2004) "The Orkut.com case: a reflection on the exploration of new ways to online sociability in the tradition of the study of virtual communities".

[5] Xiong, L., and Liu, L.,. (2003). "A reputation-based trust model for peer-to-peer ecommerce communities". IEEE Conference on E-Commerce (CEC'03). pp 275 – 281.

[6] Malaga., R. A., (2001.). "Web-based reputation management systems: Problems and suggested solutions". Electronic Commerce Research. pp 403 – 417.

[7] Gupta, M., Judge, P., and Ammar, M.,(2003) "A reputation system for peer-to-peer net works" in NOSSDAV. pp 144 -152.

[8] Damiani, E., De Capitani di Vimercati, S., Paraboschi, S.,and Samarati, P., (2003)"Managing and sharing servants'' reputations in P2P systems". IEEE Transactions on Data and Knowledge Engineering.pp 840–854.

[9] Marti, S., and Garcia-Molina, H.,(2004). "Limited reputation sharing in P2P systems". In Proc. of the 5th ACM conference on Electronic commerce, New York, NY, USA, pp 91-101,.

[10] Kamvar, S., Schlosser, M., and Garcia-Molina, H., (2003) “EigenRep: Reputation Man agement in P2P Networks”. In Proc. of ACM World Wide Web Conference, Budapest, Hungary. pp 640- 651.

[11] Damiani, E., Vimercati, S., et.al. (2002) “A Reputation-Based Approach for Choosing Re liable Resources in Peer-to-Peer Networks.” ACM Washington,DC,USA. pp 207 – 216.

[12] Nielson, S.J., Crosby, S.A., & Wallach, D.S.,(2005) “A Taxonomy of Rational Attack”. The 4th Annual International Workshop on Peer-To-Peer Systems (IPTPS 2005). pp 36 – 42

[13] Dellarocas, C., (2000) “Immunizing Online Reputation Reporting Systems Against Unfair Ratings and Discriminatory Behavior”. ACM, Minneapolis,Minnesota, USA. pp 150-157.

[14] Scott , J. J., (1969),"A chess-playing program", in Machine Intelligence 4, B. Melzer and D. Michie, Eds., Edinburgh Univ. Press, pp. 255-265.

[15] Cormen, T., Leiserson, C., Rivest, R., & Stein, C., (2001) “Introduction to Algorithms, Second Edition”. MIT Press and McGraw-Hill,. ISBN 0-262-03293-7. Section 26.2: Dijkstra’s Shortest Path Algorithm, pp.621–632.

[16] Cormen, T., Leiserson, C., Rivest, R., & Stein, C., (2001) “Introduction to Algorithms, Second Edition”. MIT Press and McGraw-Hill,. ISBN 0-262-03293-7. Section 26.2: The Ford-Fulkerson method, pp.651–664.

[17] Cormen, T., Leiserson, C., Rivest, R., & Stein, C., (2001) “Introduction to Algorithms, Second Edition”. MIT Press and McGraw-Hill,. ISBN 0-262-03293-7. Section 26.2: The Edmonds-Karp method, pp.660–663

[18] American Heritage Dictionary of the English Language,(2000) Fourth Edition.

[19] Axelrod, R., (1984) “The Evolution of Cooperation” Basic Books, ISBN 0-465-02121-2, pp 27.

[20] Dixit, A., and Nalebuff, Barry. (1991), “Thinking Strategically: The Competitive Edge in Business, Politics, and Everyday Life”. Norton, New York

[21] Turocy, T., & Stengel, B., (2001)“Game Theory”. CDAM Research Report LSE-CDAM-2001-09

[22] Guha, R., Ravi Kumar et.al (2004) “Propagation of Trust and Distrust”. ACM WWW2004 New York, New York, USA. pp 403 - 412