On Measuring Influence and its Properties in Social Networks

by

Behnam Hajian

A Thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfilment of
the requirements for the degree of

Master of Computer Science

in

Computer Science
Carleton University
Ottawa, Ontario, Canada

December 2011

Copyright ©
2011 - Behnam Hajian
Abstract

It is claimed that people prefer to get recommendations from the people they trust rather than automated advertising software. Traditional recommender systems are not capable of recognizing their users' needs at the right time due to the lack of information about their life and, on the other hand, these systems suffer from a lack of knowledge about the users’ interests. It is this lack of knowledge which has motivated research into a new generation of socially-based recommender systems. One of the issues to be resolved in social recommender systems is the identification of opinion leaders in a network. Finding effective people in societies has been a key question for politicians, marketers, security analysts, social researchers, engineers and computer scientists. The research undertaken in this thesis focuses on finding important nodes in a network based on their behaviour as well as the structure of the network. This thesis proposes an algorithm called the Probability Propagation Method for measuring the probability of infection of all the nodes in a network starting from a given node in the network. Then, assuming independence in activation of nodes in a network, a method is proposed for ranking nodes according to their capabilities in infecting a larger number of nodes in a network. These methods are validated using simulation software in which a non-deterministic model of information diffusion is simulated.
Acknowledgments

Foremost, I would like to express my sincere gratitude to my supervisor Professor Tony White. It has been an honour for me to be one of his graduate students. It was impossible to do this research without his generous guidance and support during the past two years. I have always enjoyed our weekly meeting in which he has helped me in all the aspects of this research and the writing of this thesis. I am also thankful for his financial support during the past years as well as helping me in achieving external funding support from the Mitacs organization and e-djuster. He has supported his students during the past years in terms of encouragement in their research in addition to providing cutting edge facilities in his laboratory (i.e., the Complex Adaptive System lab). I believe that I have been extremely fortunate to work with Professor Tony White as a supervisor.

I should be thankful to my family, including my parents who always encouraged me to continue my graduate study. One day I hope to complete my Ph.D., which is one of their wishes. My parents and my brothers supported me financially and emotionally for the whole of my life and I am grateful to have such a great family.
Table of Contents

Abstract ii
Acknowledgments iii
Table of Contents iv
List of Tables viii
List of Figures ix

1 Introduction 1
  1.1 Introduction 1
  1.2 Motivation 2
  1.3 Problem Statement 4
  1.4 The Coverage of the Thesis 5
  1.5 Contributions 6
  1.6 Thesis Organization 7

2 Background 8
  2.1 Introduction 8
  2.2 Definitions of a Social Network 9
  2.3 Structural Properties of a Social Network 9
  2.4 Structures in a Social Network 10
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.5 Evolution in the Networks and Referral Networks</td>
<td>42</td>
</tr>
<tr>
<td>3.6 Summary</td>
<td>45</td>
</tr>
<tr>
<td>4 Formulation of Problem</td>
<td>47</td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td>47</td>
</tr>
<tr>
<td>4.2 Definitions and Modelling of a Social Network</td>
<td>48</td>
</tr>
<tr>
<td>4.3 Modelling Information diffusion</td>
<td>51</td>
</tr>
<tr>
<td>4.3.1 Deterministic Independent Cascade Model</td>
<td>52</td>
</tr>
<tr>
<td>4.3.2 Non-deterministic Independent Cascade Model</td>
<td>52</td>
</tr>
<tr>
<td>4.3.3 Probability of Transmission</td>
<td>52</td>
</tr>
<tr>
<td>4.4 Activation Probability of a Node in a Network</td>
<td>54</td>
</tr>
<tr>
<td>4.5 Symmetric Model Vs. Asymmetric Model</td>
<td>56</td>
</tr>
<tr>
<td>4.6 Influence Maximization Problems</td>
<td>56</td>
</tr>
<tr>
<td>4.6.1 Measuring Influence in a Social Network</td>
<td>57</td>
</tr>
<tr>
<td>4.6.2 Influence Maximization Problem</td>
<td>57</td>
</tr>
<tr>
<td>4.7 Measuring Influence in the Deterministic Cascade Model</td>
<td>58</td>
</tr>
<tr>
<td>4.8 Influence Maximization in the Symmetric Deterministic Cascade Model</td>
<td>59</td>
</tr>
<tr>
<td>4.9 Influence Maximization in Asymmetric Deterministic Cascade Model</td>
<td>63</td>
</tr>
<tr>
<td>4.10 Measuring Influence in non-Deterministic Cascade Model</td>
<td>65</td>
</tr>
<tr>
<td>4.11 Summary</td>
<td>66</td>
</tr>
<tr>
<td>5 Influence Measurement Algorithms</td>
<td>68</td>
</tr>
<tr>
<td>5.1 Introduction</td>
<td>68</td>
</tr>
<tr>
<td>5.2 Measuring Probability of Activation in Non-deterministic Model</td>
<td>69</td>
</tr>
<tr>
<td>5.2.1 Modelling Activation Process by Markov Chain Model</td>
<td>69</td>
</tr>
<tr>
<td>5.2.2 Approximating Activation probability by Iteration</td>
<td>71</td>
</tr>
<tr>
<td>5.2.3 Probability Propagation Algorithm</td>
<td>72</td>
</tr>
<tr>
<td>5.2.4 Convergence of The method</td>
<td>73</td>
</tr>
</tbody>
</table>
## List of Tables

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Correlation between pairs of Influence metrics in Meeyoung Cha et. al. research</td>
<td>24</td>
</tr>
<tr>
<td>2</td>
<td>Yu and Singh Trust model update rules</td>
<td>39</td>
</tr>
<tr>
<td>3</td>
<td>Structural properties of Epinion and Facebook datasets</td>
<td>85</td>
</tr>
<tr>
<td>4</td>
<td>Properties of the networks used in the simulation. The last column shows the probability of existence of an edge in a Random-Graph or clustering co-efficient in a small-world network according to which random graphs or small-world networks were generated.</td>
<td>86</td>
</tr>
<tr>
<td>5</td>
<td>Statistics about the maximum number and level of propagations and hops to which a posting penetrated</td>
<td>87</td>
</tr>
<tr>
<td>6</td>
<td>Statistics about the average number and level of propagations and hops to which a posting penetrated</td>
<td>88</td>
</tr>
<tr>
<td>7</td>
<td>Correlation between anticipated number of infected nodes by the proposed algorithm and results of simulation; the last column shows correlation between probability of activation of nodes in a network and probability of activation for the nodes in the simulation</td>
<td>89</td>
</tr>
</tbody>
</table>
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Advertisement bombardment Vs. social recommendation</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Viral Advertising versus traditional advertising</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>Transitive Triple vs. closed triple</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>Transition Chart in Markov Chain Model</td>
<td>17</td>
</tr>
<tr>
<td>5</td>
<td>Distribution of diffusion of innovation [50]</td>
<td>26</td>
</tr>
<tr>
<td>6</td>
<td>Calculating similarity using Multi-tree model</td>
<td>35</td>
</tr>
<tr>
<td>7</td>
<td>Transmission probability of infection of a vertex $v_j$ from its neighbour $v_i$</td>
<td>54</td>
</tr>
<tr>
<td>8</td>
<td>Multi-path transmission in a network</td>
<td>55</td>
</tr>
<tr>
<td>9</td>
<td>Example of applying of BFS on a graph in a non-deterministic symmetric model; the threshold $\theta_{v_i}$ for all the nodes are considered equal to 0.6</td>
<td>61</td>
</tr>
<tr>
<td>10</td>
<td>Diffusion in Deterministic Cascade Model</td>
<td>63</td>
</tr>
<tr>
<td>11</td>
<td>Overlap in the influence sets</td>
<td>64</td>
</tr>
<tr>
<td>12</td>
<td>Modelling activations with Markov Chain Model</td>
<td>70</td>
</tr>
<tr>
<td>13</td>
<td>Cycles in a network causes the probabilities to converge to 1 by propagation of probabilities</td>
<td>73</td>
</tr>
<tr>
<td>14</td>
<td>Removing cycles by eliminating outgoing edges in the graph for the nodes a,b,c,d from left to right</td>
<td>75</td>
</tr>
<tr>
<td>Page</td>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>------</td>
<td>--------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>15</td>
<td>Assuming a direct edge from the starting node to all the nodes in the</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>graph</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Normal Distribution [82]</td>
<td>81</td>
</tr>
<tr>
<td>17</td>
<td>Correlation between probability in simulation and probability propagation algorithm for 10 and 100 nodes</td>
<td>91</td>
</tr>
<tr>
<td>18</td>
<td>Correlation between probability in simulation and probability propagation algorithm for 1000 nodes</td>
<td>92</td>
</tr>
<tr>
<td>19</td>
<td>Relation between number of nodes and number of edges in different networks and maximum degree in these networks</td>
<td>93</td>
</tr>
<tr>
<td>20</td>
<td>Average number of infected nodes, average number of hops</td>
<td>94</td>
</tr>
<tr>
<td>21</td>
<td>Relation between Spearman’s correlation and number of nodes</td>
<td>95</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Introduction

Finding effective people in societies has been a key question for politicians, marketers, security analysts, social researchers, engineers and computer scientists. Since any society can be considered as a network, network analysis has provided significant insight in this area. This question has also been of interest to physicians and the companies which produce firewalls and security software for computer viruses, malware and computer worms. Social and economic networks have been studied for decades in order to mine useful information not only to organize these networks in a way that maximum efficiency is accomplished but also to understand the role of nodes or groups in a network [1,2]. This research focuses on finding important nodes in a network based on their behaviour as well as the structure of the network. By the emergence of social networks in recent years, finding influential nodes has absorbed a considerable amount of attention from researchers in this area [3,4].
1.2 Motivation

On-line recommender systems suffer from a lack of attention from their users generally because their users feel suggestions from these systems are useless [5]. However, it has been proven that the main reason for this feeling is that people prefer to get recommendations from the people they trust rather than automated advertising software [6]. On the one hand, these systems are not capable of recognizing their users’ needs at the right time due to the lack of information about their life and, on the other hand, these systems suffer from a lack of knowledge about the users’ interests [7]. The former problem has not been studied extensively and the latter is generally called the Cold Start Problem [5] in on-line recommender systems literature. The former problem above can be stated as a lack of knowledge about a period of time in which a person may consider buying a specific service or a product. As an example, young people change their laptops or their cellphones every 2 years to keep up with new technology. Therefore, it would be bothersome for a person to get recommendations about laptops when he/she recently bought a new laptop. This fact depends on the interest, lifestyle, wealth of their family, imitation behaviour, age and availability of funds for a customer to be in a situation in which an advertisement makes him purchase a new product. However, if these factors are ignored by marketers and their advertisements are not delivered to the user at the right time, their advertisement may be ignored by the customer.

According to Iyengar et al. [8], friends have a significant effect on the purchase probability of a customer for a specific product. These researchers discovered that the social effect is zero for 48% of social network users, negative for 12% of the users and positive for 40% of the users. On the one hand, imitation is also effective in increasing revenue and sales, while users tend to buy some of the new products when their friends already bought that product. On the other hand, sometimes this fact has
a negative effect, since some users are reluctant to buy a product which has already been bought by their friends.

Figure 1: Advertisement bombardment Vs. social recommendation

Adaptability to innovations and new products is another issue specifically for marketing cutting edge products. However, it has been observed that people are infected by a new idea from being exposed to the innovation by their friends rather than advertisements. Figure 1 visualizes this observation that getting ad hoc advertisements is annoying and inefficient in maintaining customers’ attention. As an example, there is higher probability for buying a product from a new user if he/she works with a touch screen tablet bought by his/her friend than watching an advertisement about the same product in his email or on the street. This observation is shown in Figure 2. This fact motivates marketers to turn to viral marketing rather than ad-hoc advertising.

A key question in viral marketing is finding people who have more influence on their society and can affect more people in their community and those people can affect more people recursively [9]. In viral marketing, the most influential people
are identified and the new product (e.g., iPhone 5) is sold to them with a considerable discount and they advertise the product themselves. This type of recommender systems is sometimes referred to as *social recommender system*.

### 1.3 Problem Statement

This thesis focuses on how to find the best nodes that if those nodes are infected with an idea, they can infect more nodes than their rivals. This problem is called influence maximization because influence is modelled as the ability of a node to infect more nodes in the network. Here, two questions are posed: First, how to rank nodes based on their importance of infecting more nodes if that node individually gets infected. Second, how to construct a set of nodes with a given size to maximize the number of infected nodes in a network. The first problem focuses on the effect of each individual on the infection of nodes in the network. However, in the second problem, the influence of an individual is not important by itself, but we focus on the combination of nodes in a set and we expect an increasing number of nodes get

---

**Figure 2:** Viral Advertising versus traditional advertising
infected by adding a new node to the set. In the second problem, we have to consider
overlaps among sets of infected nodes. In other words, while we try to find the node
with the maximum number of infections, the overlap between the new set and the set
of already infected nodes has to be minimized. These problems are formally described
in Chapter 4.

1.4 The Coverage of the Thesis

This thesis proposes a new mechanism for recommending products to on-line cus-
tomers. According to the current trend in moving marketers toward social recom-
mender systems and the difficulties in finding influential people in this environment,
the influence maximization problem in social networks is considered as a controversial
problem. Two different categories of problems arise in this area. These problems are
formally defined in this thesis. The first problem deals with infecting of nodes with
a new idea by starting from a subset of initial nodes. The included nodes in the
initial subset are not necessarily the most influential nodes in the network, but the
combination of those nodes is important in infecting greater number of nodes in the
given network. The second problem deals with ranking individuals according to their
abilities in infecting more nodes in the network. This rank reflects the influence of a
user in the network. The second problem is referred to as measuring influence in a
network.

These two problems are formally defined in Chapter 4 but the first problem is
not the problem of primary interest of this thesis. However, the goal of this thesis
is measuring influence as described above. In this thesis, both problems are defined
formally, and NP-completeness of finding the best combination of nodes is described.
Furthermore, this thesis proposes an algorithm to measure the influence of a node in
a given network according to the number of infected nodes starting from that node.
However, this thesis does not propose a solution that maximize infection. This second problem acknowledge that different nodes may infect same set of nodes.

1.5 Contributions

This thesis provides 4 contributions enumerated as follow. First, a social network is formally represented. While influence in social networks can be viewed in an abstract way, dealing with heterogeneous relationships and interactions, we focus on modelling a specific social network which is extendible to a general framework for any type of relationship and interaction among individuals.

Second, this framework has been implemented and simulated for validation of results. The general framework may stand as a network of viruses, marketing network or a social network such as Facebook, Flicker or Twitter.

The Third contribution of this thesis is the determination of the problem of maximizing the number of infected nodes in the independent cascade models. The deterministic model versus non-deterministic model of diffusion of information in a network are described. Symmetric model versus asymmetric model of information diffusion is also described. The complexity of algorithms for the mentioned problems in Section 1.3 is also described.

This thesis focuses on a probabilistic model of diffusion of information in social networks. Two models are proposed to represent the behaviour of individuals in a social network in terms of propagation of information. The forth contribution is computing the expected number of infected nodes resulting from a posting in a social network given the starting point is the main goal of this research. This problem is formulated by considering a random process for propagation of information in a network. Furthermore, an algorithm for measuring the probability of infection of a node starting from a particular node in a network is proposed. These probabilities
are used to compute the expected number of infections in a network. The proposed algorithms are validated using simulation of propagation of information in the independent cascade model.

1.6 Thesis Organization

This thesis is organized in seven chapters. In this chapter, the introduction, the problem statement and contribution were discussed. Chapter two includes background information about the subject area, including social networks and trust. This chapter includes definitions central to the material described in Chapter 3 and beyond. In Chapter three, related work is analyzed in preparation for models and algorithms which are proposed in Chapter 4 and 5. The literature analyzed in Chapter 3 also covers a number of sub-problems used for the simulation and potential applications of this research. Chapter four discusses the problem of maximization of influence in a social network and the modelling of behaviours in this environment. Chapter five provides solutions and describes proposed algorithms for measuring influence in a social network. In order to validate the proposed algorithms in this thesis, a general framework of a social network is simulated. Chapter six discusses the properties and details of our simulations, experimental results and includes a discussion of the results. Finally, conclusions and future work will be discussed in Chapter seven.
Chapter 2

Background

2.1 Introduction

Previous literature [10] has defined the phrase social network as a combination of individuals represented by nodes and links between pairs of nodes forming the structure of a network. A small number of papers concentrate on the content analysis and behaviour of individuals in a network as well as structural analysis in a network. This research focuses on the behaviour of individuals in a network as well as structural analysis of the corresponding network.

In order to analyze any complex system, the system needs to be formally defined [1]. In this thesis a social network is a complex system and in order to understand the behaviour of individuals in the network, the network, its structure and its properties need to be determined and formally expressed. This section provides basic information about social networks as well as background information about what we need in the following chapters. These properties and behaviours will be discussed in more detail in Chapter five. As discussed in the previous chapter, the problem studied in this thesis is the measurement of influence in a social network. Therefore, the present chapter contains a set of definitions such as the meaning of influence, definition of a network, diffusion model, trust, similarity, some graph theory and finally
Markov chain process as a background for the suggested solution in Chapter 6.

2.2 Definitions of a Social Network

The term *social network* is generally defined as a set of individuals (or organizations) which are connected together with different relationships such as friendship, sexual relationship, financial exchange, dislike, same beliefs or any commonality that ties two individuals together [2, 10]. The rest of this section is devoted to the formal definition of a social network. However, the definition of a social network in this paragraph is more general while in the following sections, the defined social network is more specific for the domain of discourse of this thesis.

2.3 Structural Properties of a Social Network

In order to analyse a network based on structure, several factors are defined in the following paragraphs such as degree distribution, clustering, network density and diameter of a network.

**Definition** *Node degree* is the number of edges which start from the node and connect the node with other nodes in the network. *Degree distribution* is a fundamental characteristic of a network which is defined as the relative frequencies of nodes with different degrees. For example, a regular network is the one in which all the nodes have the same degree [1].

**Definition** *Network Density* is the relative fraction of the links present in a network. This parameter can be simply measured by dividing average node degree by \( n - 1 \) [1].

**Definition** According to Jackson [1], a maximal completely connected sub-network in a given network is called *clique*. Similarly, *Cliquishness* is a property of a network which is defined according to the number and size of cliques in the network [1].
**Definition** Since by removing a link in network, the clique structure will completely change, we say that clique structure is said to be very sensitive to the structure of a network. As a result, we need another metric for measuring cliquishness. *Clustering coefficient* is defined based on the transitive triples as the fraction of number of closed triples to the number of connected triples demonstrated in Figure 3. The following equation refers to clustering coefficient where \( \eta(i) \) returns the set of immediate neighbourhoods for the vertex \( v_i \) [1].

\[
    Cl(G) = \frac{\sum_i | \{<j,k> \in E | k \neq j, j \in \eta(i), k \in \eta(i)\} |}{\sum_i | \{<j,k>| k \neq j, j \in \eta(i), k \in \eta(i)\} |},
\]

(1)

**Figure 3:** Transitive Triple vs. closed triple

**Definition** The number of links between two nodes is called the *distance* between two nodes. The *diameter* of a network is defined as the largest distance between any pairs of nodes in the network. Since diameter captures the upper bound of path length in a network, the *average path length* is another metric used to measure the clustering coefficient of a network [1].

### 2.4 Structures in a Social Network

Different network structures (a.k.a., topologies) are created according to different views of the network, different types of users and the nature of the social network. As an example, a network can be created according to the probability of existence of a
link between two nodes or by following a specific distribution for degree of each node in the graph [1,10]. In this section, different models of network structures are studied including: Regular Network, Scale-free Network, Random Network, and Small World Network.

**Definition** A regular network is the one in which the degree of all vertices are equal. Therefore, in a regular network $\forall d \neq k : P(k) = 1$ and $P(d) = 0$ [1] where $P(d)$ refers to the probability of existence of a node with degree $d$.

**Definition** A random network is an structure in which the degree of a node follows a Poisson distribution. Hence, the formation of nodes of specific degree in a network follows the binomial expression [1,11]. In random networks, the probability that any given node $i$ has exactly $d$ links is

$$\binom{n-1}{d}p^d(1-p)^{n-1-d}$$

(2)

For large $n$ and small $p$, this binomial expression is approximated by a Poisson distribution and, as a result, the fraction of nodes that have $d$ links is approximately [1]:

$$\frac{e^{-(n-1)p}((n-1)p)^d}{d!}$$

(3)

**Definition** In a scale-free network, nodes’ degrees satisfy a power law distribution as shown in the following equation:

$$P(d) = cd^{-\gamma}$$

(4)

The above equation means, if the degree is increased by factor $k$ then the frequency will be reduced by $k^{-\gamma}$. The reason that these networks are called scale-free is that the relative probability of degrees of a fixed ratio are equivalent, independent of the scale.
of those degrees [1]. For instance, in a scale free network, \( P(5)/P(1) = P(50)/P(10) \). The highest degree nodes in these graphs are called hubs.

This topology is widely found in natural and artificial systems. It was formally defined by Albert-Laszlo Barabasi [12]. For example, Derek de Solla Price showed that the number of citations in scientific papers follows the power law distribution.

**Definition** Another type of network structure is a small world network [13] in which most of the nodes are reachable in a small number of hops while most of the nodes are not neighbourhoods of one another. In some literature, a small world network is defined as a network in which the average distance between two randomly chosen nodes, \( \text{dist}(v_n, v_m) \), is proportional to the logarithm of the number of nodes in the corresponding network [1]. Formally,

\[
\text{dist}(v_n, v_m) \propto \log(|V|) \tag{5}
\]

According to Duncan Watts and Steven Strogatz [13], any graph can be classified by clustering coefficient and average node-to-node distance. A small world network also tends to contain cliques.

### 2.5 Definition of Influence in Social Network

In Webster’s dictionary the word *influence* is defined as, "The power or capacity of a person or things in causing an effect in indirect or intangible ways". The study of influence dates back over a century. The role of influence and its effects is studied extensively in sociology [4, 14], communication [10], marketing and political science and in understanding peer pressure, obedience and leadership. Three broad categories of social influence were introduced by Hernert Kelman [15]: (i) Compliance, which is defined as agreement among people by keeping the difference of their opinions private.
(ii) Identification, in which people are being influenced by someone who is liked and respected, such as a famous celebrity. (iii) Internalization, in which people accept a belief or behaviour and agreement is made publicly and privately. Generally, social influence is a phenomenon in which the actions of a user cause his/her friends to behave in a similar way.

In this thesis, influence is considered to be the ability of a person to convince others to behave in a similar way. This characteristic depends on several factors, such as: charisma, reputation, social personality and psychological manipulation abilities. In small communities in a social network the behaviour of each person is important in influencing his friends or followers. For example, the high number of postings in social networks such as Facebook decreases attention to ones postings [16]. On the one hand, the quality of a posting is important for the followers of a specific user as well as the subject of a posting. A subject might be interesting for a group of followers while it is bothersome or boring for another group of people. On the other hand, the quality of the personality of a person is another aspect of being influential in a network. The latter is more commonly known as trust which is established between two individuals [17]. The other factor of influencing someone is commonality of interests which is referred as similarity in this thesis.

2.6 Diffusion Models in Social Network

Diffusion of information is the process of spreading of information through the members of a social network [18, 19]. An example of this process in a social network is re-sharing of postings in a social network. Scientists model this process formally in order to analyze, maximize or minimize it. As this process is similar to diffusion in physical phenomena, information diffusion has been modelled by physical diffusion models [20]. Other researchers modelled this phenomenon in the network from a
sociological point of view. There are several mathematical models that have been suggested to formally model the spread of information in the network [21]. However, none of those is comprehensive because the nature of this process depends on the model of the nature of the network. Hence, this process needs to be formalized according to the model of the corresponding network. This process is formally defined later in Chapter 3.

2.7 Trust

Trust is the impression of an individual about his/her friends or in other words about other individuals with whom he/she has interacted [22–24]. This impression is topical and refers to the expertise of its friends [25]. Reputation is the evaluation of the quality of the characteristic of an individual based on a social judgement [22]. Trust and reputation are important issues in many different domains and applications from marketing to learning in multi agent systems [25] and P2P file sharing systems [26]. However, many different models have been suggested by researchers in many different domains that utilize trust. Some of these models are analyzed later in Chapter 3. In this research, trust between two individuals is considered as one of the factors in convincing an individual in a social network to behave in a similar way. In fact, one of the determining factors in spreading information from one node to another one is trust and belief in the correctness of the information that an individual gets from its neighbourhoods [27]. In viral marketing and social recommender systems, trust is an essential factor used to encourage a user to buy a new product according to the information that is obtained from one of the trusted neighbours. These models are discussed in Section 3.4.
2.8 Similarity

Similarity is a key issue in search engines, information retrieval systems, and recommender systems. Similarity refers to the psychological nearness of two concepts [28]. Similarity has roots in psychology, social sciences, mathematics, physics and computer science. In social psychology, similarity points to how closely attitudes, values, interests and personality match between people which can lead to interpersonal attraction. This can be explained by the fact that similar people tend to place themselves in similar settings and this consequently decreases potential conflicts between them. Furthermore, finding a person with similar tastes helps to validate values or views held in common. With a mental representation, the similarity between two concepts is defined as a function of the distance between two concepts represented as different points in the mental space [29].

Conventionally, statistical means can estimate the distance between two entities by comparing features representing entities [30]. Simply put, concepts which are near to each other are more similar than points which are conceptually distant. There are several mathematical techniques for estimating this distance, such as Latent Semantic Analysis (LSA) [31]. In order to measure similarity between two entities, each entity can be represented by a vector of its features. For example, \( f_{ij} \) in the vector \( e_j = (f_{1j}, ..., f_{Fj}) \) represents feature \( i \) of entity \( j \) where \( F \) represents the total number of features in the vector [30]. This method of representing entities is called the Vector Space Model (VSM). Each \( f_{ij} \) can be represented as a binary value or a real number indicating the degree of relatedness of the feature to the entity. In Chapter 3, another model is suggested and then advantages and disadvantages of these models are discussed in more detail.
2.9 Graph Theory Definitions

The study of graphs and mathematical structure for modelling relations between entities is called graph theory. Here, an important problem is discussed in the following section which is essential material in proving NP-completeness of the influence maximization problem in Chapter 4.

2.9.1 Maximum Clique problem

Finding a complete sub-graph (a.k.a., clique) in a graph is called the maximum clique problem [32]. This problem is also known as finding sets of elements where each pair is mutually connected while the problem can be reduced to the set covering problem.

The set covering problem is a classical problem in complexity theory of computing [33] which is shown to be a NP-complete. Later, in Chapter 4, we try to find a smallest set of sub-sets of objects whose union includes all the elements in the universe. More formally, we have $\mathcal{U}$ as the universe of objects and $\mathcal{S}$ as the set of sub-sets of $\mathcal{U}$. A cover set is the set $\mathcal{C} \subseteq \mathcal{U}$ whose union is equal to $\mathcal{U}$. A set covering problem is a task of finding a set covering using the fewest number of sets given the input pair $(\mathcal{U}, \mathcal{S})$. The problem is NP-complete because finding the set $\mathcal{C}$ is non-polynomial but verification of the existence of $\mathcal{C}$ as the answer is a polynomial problem [33].

The problem of finding a clique with the largest number of vertices is NP-complete which is reducible to the cover set problem while the brute force algorithm to find the maximum clique problem has exponential time complexity.

2.10 Markov Chain Theory

In a Markov chain process, the outcome of a given experiment may affect future experiments. In his definition, a tree can represent a sequence of $n$ experiments when
a sequence of random experiments forms an independent trial process. Furthermore, the knowledge of outcomes of previous experiments does not affect the prediction for the outcomes of the next experiment [34].

Markov chain can be described as a set of states \( S = \{s_1, s_2, ..., s_n\} \) and the process begins from one of the states and moves successively from one state to another one. When the chain is in the state \( s_i \) then it moves to the next state \( s_j \) with the transition probability \( p_{ij} \) which is independent of the chain of steps it followed before the current state [34]. Figure 4 demonstrates a state diagram representing a Markov chain process. The Markov chain can be represented as the sequence of jumps in which a frog jumps from one lily pad to another with an appropriate transition probability. This concept will be used in our algorithm to predict the influence of a vertex in a social network according to the probability of re-sharing of a posting by the adjacent vertices.

Figure 4: Transition Chart in Markov Chain Model
2.11 Summary

This chapter provided some background information about what we need to describe the problem of measuring influence in social networks. Therefore, essential definitions such as definition for social influence, different network structures, trust, similarity and information diffusion were described. Following this, the academic definition for influence was provided and finally some mathematical theories were discussed which are useful in the following chapters to formalize the algorithms proposed in this thesis. Markov chain theory forms the basis for our suggested algorithm. The next chapter discusses related work and concentrates in more detail on formal aspects of the material introduced in this chapter.
Chapter 3

Related Work

3.1 Introduction

Social Networks are infrastructures for business, advertising, innovation propagation, news feeds, communicating among groups and friends, and many other utilities. For more than a century, the term Social Network has been used in different domains, such as economics, social sciences and computer science, to denote the notion of patterns and ties among members of a society or a community [2]. In other words, a social network is a network of social ties which is modelled by a graph constructed from nodes and edges between nodes representing individuals and relations between individuals in a society respectively [1]. In different networks, these ties represent different concepts from friendships and sexual relationships to physical links between two routers or two computers in telecommunication networks.

In Social Network Analysis (SNA), we concentrate on the mapping of relations and flows between users, groups, organizations, computers, people or any other entities, to a formal model which facilitates understanding of how behaviour of users is affected by the structures of a network as well as which network structure is more likely to emerge [10]. Network analysis has found applications in many domains beyond social sciences [4]; for example, the study of food chains in different ecosystems [3],
identifying criminal and terrorist networks from traces of collected communications [4] and understanding the interaction of proteins in metabolic pathways [35]. Most recently, the intersection of control theory and network analysis has led to a greater understanding of how control of complex networks might be achieved [36].

3.2 Social Network Mining and Analysis

Network Analysis has been studied in several domains such as identifying terrorist networks, optimizing wireless and cable network operators and epidemiology. SNA helps us to understand how contacts among humans could affect the spread of diseases and rumours in human networks as well as designing strategies for preventing the spread of computer worms in a network [3]. In order to identify the roles of individuals in the network, SNA evaluates the location of actors in the network as well as analyzing their behaviours and their interactions. We distinguish two classes of network analysis as follows: Structural Analysis which concentrates on measuring centrality of users and defining their roles; Behavioural Analysis which concentrates on the interactions between users rather than the structure of the network [37].

A fundamental question in network analysis is finding relatively important nodes in propagating information through the network. In order to answer this question, we think about metrics for evaluating the importance and the role of each node in the network [37]. In the following sections, we define several metrics for evaluating centrality of a user according to the structure of a network.

3.2.1 Structural Analysis Metrics

Definition Degree Centrality is the simplest centrality metric. Degree centrality is defined as the ratio of the degree of the node divided by $m - 1$, where $m$ is the number of nodes in the network. This metric is interpreted as the risk of being infection for
a node from its immediate neighbourhoods [1,38,39]. We distinguish between in-degree and out-degree when a network is directed. In-degree is often interpreted as popularity and out-degree is interpreted as sociability.

\[ C_D(v) = \frac{\text{deg}(v)}{(m - 1)}, \]  

(6) where \( \text{deg}(v) \) is the degree of the node \( v \).

**Definition** *Closeness Centrality* is one of the determining measures of importance of a node in a network [1,39]. This metric is defined as the mean length of all shortest paths from a node to all the other nodes in the network. Since it is computationally hard to calculate this metric, several algorithms using random walk centrality are used to calculate this metric. According to Freeman, this metric can be calculated using following equation [40]:

\[ C_C = \frac{\sum_{t \in V \setminus v} d_G(v,t)}{n - 1}, \]  

(7) Where \( d_G(v,t) \) is the distance of the node \( v \) from the node \( t \) in the graph \( G \). A node with higher closeness centrality has quicker accessibility to the entire network and, as a consequence, is more visible in the network. Hence, nodes with a low value of closeness score are tend to receive more flows. These nodes are assumed to be well positioned to obtain novel information earlier than others. On the other hand, these nodes are expected to catch infections sooner in case of new diseases [40].

**Definition** *Betweenness Centrality* is another centrality measure introduced by Freeman [1,39,41,42]. This metric is defined as the relative number of shortest paths that pass through a node for which the betweenness is measured from all the vertices to all the others to all the shortest paths in the network. Similarly, the betweenness metric can be defined regarding the edge betweenness centrality [42]. The betweenness centrality of a node \( k \) is given by:
The betweenness centrality measures the amount of network flow that a given node controls while this measure gives the volume of the traffic that passes through a node in the network. On the other hand, this metric can measure the possibility of infection by the traffic that passes through a node as well as control over diffusion of infection to other nodes in the network [42]. However, this metric makes sense first when the traffic travels only through a shortest path rather than diffusing randomly from a node to its neighbourhood. Secondly, the traffic should randomly choose one of the shortest paths among the others. Therefore, this metric ignores issues such as trust between two nodes as the weight of their relationship. A fundamental question here is what kind of flow follows these rules? The answer to this question simply ignores the importance of betweenness centrality in the spread of infection or information throughout a network [41]. Also, gossip does not have these properties while such information is replicated rather than moving through a network and usually does not have a target. The calculation of betweenness centrality is proven to be NP-Hard by Dolev et al. and can be approximated by \((1 - 1/e)\) using a greedy algorithm [43].

**Definition** *Eigenvector Centrality* is a measure of popularity of a node in a network. A node’s popularity is proportional to the sum of popularity of all nodes directly connected to it [39, 40]. This metric is recursively calculated in the network according to the value of other nodes’ eigenvector centrality values. This metric is used to identify nodes which are connected to the well-connected nodes. The Google-PageRank algorithm [44, 45] is used to approximate Eigenvector centrality. The actual value of eigenvector centrality may be calculated in non-polynomial time and is often estimated by algorithms such as Google PageRank [44, 46]. The Eigenvector centrality is calculated by the following equation in which \(x_i\) is the score of the \(i^{th}\) node and
$A = (a_{ij})$ is the adjacency matrix of the network. In this equation, the eigenvector of a node is proportional to the sum of eigenvector centrality of its neighbourhoods:

$$x_i = \frac{1}{\lambda} \sum_{j \in N(i)} x_j = \frac{1}{\lambda} \sum_{j=1}^{n} a_{ij} x_j.$$ \hspace{1cm} (9)

In this equation, $N(i)$ is the set of all the neighbourhoods of the node $i$ and $N$ is the total number of nodes in the network. This equation can be rewritten as $Ax = \lambda x$ [39].

### 3.2.2 Behavioural Analysis of a network

The metrics described above concentrate on the analysis of the structure of a network rather than the behaviour of nodes and their interactions. However, some researchers take other aspects of analysis – such as users’ behaviours – into consideration. In some research, a weight is assigned to each tie showing importance and strength of a link between two users based on their historical interactions. Some of these works are discussed in the following paragraphs.

The measurement of influence in Twitter has been studied by Meeyoung Cha et al. based on the following metrics [47]:

- **In-degree**: is the number of followers of a user. In-degree represents the popularity of a user in a social network.

- **Retweets**: which is defined as the number of times that followers of a specific node pass along a posting from a tweeter. Retweeting causes propagation of a posting or news in a network. This metric is important as it shows how an advertisement can propagate across the network using influential users.

- **Mentions**: this means the number of times that the name of a user is mentioned in his followers’ postings. This metric has been observed to follow a power law
distribution.

These researchers used Spearman’s rank correlation coefficient for comparing correlation between each pair of the above ranks for all the users in Twitter. The research compared the three measures mentioned above to analyze topics of the most influential people in Twitter according to the aforementioned analysis and retweets. This research also revealed that influence is not gained accidentally but requires that users need to be consistently active in the network. The following table shows their results in comparing pairs of the aforementioned metrics for influence [47]:

<table>
<thead>
<tr>
<th>Correlation</th>
<th>All</th>
<th>Top 10</th>
<th>Top 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indegree vs retweets</td>
<td>0.549</td>
<td>0.122</td>
<td>0.109</td>
</tr>
<tr>
<td>Indegree vs mentions</td>
<td>0.638</td>
<td>0.286</td>
<td>0.309</td>
</tr>
<tr>
<td>Retweets vs mentions</td>
<td>0.580</td>
<td>0.638</td>
<td>0.605</td>
</tr>
</tbody>
</table>

Table 1: Correlation between pairs of Influence metrics in Meeyoung Cha et. al. research

Meeyoung Cha et al. also studied the dynamics of influence over topic and time and revealed that most influential users can maintain a significant influence on different topics. They have noted that new organizations are shown to be influential in terms of retweeting contents [47].

Dynamic Graph Analysis has been studied by Khrabrov and Cybenko [48]. Here, the number of daily mentions for each user is considered as a indicator for computing different ranks for influence analysis of each node in a network. They studied the effect of PageRank, drank, and starrank on a Twitter dataset as metrics to study influence in the network over time. Their notion of PageRank is the same as the Google PageRank concept computed over the number of mentions for each user. They introduced drank as the row number of users sorted by their PageRank, dirank, drrank and starrank with normalization on number of tweets over time. These researchers
used several primitive indices in combination, such as Contiguous Longest Increasing Sub-sequences (CLIS) and GrowFall for analysis of influence ranks during a period of time. These indices show how the influence rank of a user changes with time.

Khrabrov and Cybenko also analyzed the rate of increase in the number of mentions for influencing users in a network for consecutive days. These researchers also insisted that in order to be influential, a user needs to be consistently active in the network [48].

Afrasiabi and Benyoucef studied influence as a combination of link strength and incoming/outgoing clustering value defined for each node in the network [16]. The link strength is measured according to the volume of interactions among users while the clustering value is measured by the closeness of a node to highly interconnected communities. This approach has been applied on a YouTube dataset collection crawled from the Internet. They filter the spam and inactive nodes according to their activities and their interaction with other users. According to their research, it is revealed that users with a high value of content generation rate which have strong relationship with their neighborhoods and are close to their cluster can contribute significantly for propagation of content in their community. They studied propagation of information on an open social network dataset (i.e., YouTube) over subscription in addition to friendships in this network [16].

### 3.2.3 Information Diffusion Models in Social Networks

A diffusion process can be used to describe the information flows in social media [1]. This research has been ongoing since the publication of a book called "Diffusion of Innovations" by Rogers [49].

**Bass Model**

Rogers [49] categorizes the adopters of any new innovation into innovators (2.5), early adopters (13.5), early majority (34), late majority (34) and laggards (16). These
groups are shown in Figure 5.

**Figure 5:** Distribution of diffusion of innovation [50]

In another approach, Frank Bass formulated the characterization of spread of a new product or a technology in a market by the following equation, in that $F(t)$ is the fraction of agents which adopt a new product or behaviour in a discrete time period $t$ [1,49]:

$$F(t) = F(t-1) + p(1 - F(t-1)) + q(1 - F(t-1))F(t-1), \quad (10)$$

In the above equation, $p$ is the rate of innovation and $q$ is that of imitation. The expression $p(1 - F(t-1))$ captures the rate of innovation times not-adopted fraction of the agents, while $q(1 - F(t-1))F(t-1)$ is the imitating fraction of diffusion process. Solving the above equation and setting $F(0) = 0$ leads to [1,49]:

$$F(t) = \frac{1 - e^{-(p+q)t}}{1 + (\frac{q}{p}e^{-(p+q)t})}, \quad (11)$$
This model is capable of capturing a wide variety of diffusion curves. Hence, this model can be used in either forecasting or empirical analysis of a diffusion process. This model can fit into studies such as Ryan and Gross [51] who emphasized on finding diffusion curves that are S-shaped. The S-shape corroborates that diffusion begins slowly, then accelerates, slows down and finally collapses [1].

**Linear Influence Model**

Jaewon Yang and Jure Leskovec address the infection process of nodes in networks by modelling diffusion without any need for explicit knowledge of the network [19]. In their model, they ignore the fact of which node infects which other nodes, but they focus on the rate of infection through the implicit network. In their research, they focus on the rate of diffusion in which spread of information either decays or rises. Jaewon and Jure define the Linear Influence Model (LIM) [19] by starting with the assumption that the number of newly infected nodes depends on the number of infected nodes in the past. The number of infected nodes has been formulated by their model as a function of time. Modelling the number of infected nodes as a function of time given the number of previously infected nodes, the researchers determined that the influence function developed into different shapes according to the node’s type (i.e., individual, news paper, blog, etc.). The diffusion process studied in this paper revealed patterns of influence in different types of contagions. This fact forces us to take topic of product or the topic of the information into consideration for the diffusion process.

Jaewon and Jure defined the volume $V(t)$ as the number of nodes that mention the information at time $t$ as a function of which node mentioned the information beforehand. This function is defined as follows, where $I_u(l)$ is the influence function of node $u$ returning number of follow up $l$ time after $u$ adapted the information [19].
\[ V(t+1) = \sum_{u \in A(t)} I_u(t - t_u), \quad (12) \]

\( A(t) \) denotes the set of already active nodes that were activated prior to time \( t \) \((t_u < t)\). The individual influence function can also be modelled by either an exponential \( I_u(l) = e^{-\lambda u l} \) or a power law \( I_u(l) = c_u l^{-\alpha_u} \). The above non-parametric formulation of LIM does not consider the shape of the individual influence function.

The extension of LIM in which a function, \( M_{u,k}(t-l) \) indicating infection of node \( u \) by \( k \) at time \( t \) is added, which is defined as follows \([19]\):

\[ V_k(t+1) = \sum_{u=1}^{u=N} \sum_{l=0}^{L-1} M_{u,k}(t-l) I_u(l + 1), \quad (13) \]

Finally, the researchers defined the \( \alpha - LIM \) and \( \beta - LIM \) models which consider novelty and imitation respectively. According to the reported experimental results the researchers showed that LIM outperforms mathematical time series in prediction over 170 million news media and 500 million tweets. Furthermore, it is observed that imitation and novelty have a strong strength on the adoption of short textual phrases in online news media \([19]\).

**Heat Diffusion Model**

Another work in this area has modelled a social network by a heat diffusion process for marketing candidates selection \([20]\). Heat diffusion is a physical phenomenon in which heat flows from a high temperature region to the lower one. In this research, heat kernel models the information flow in networks. Hao Ma et al. use a second order differential equation to describe the information flow in the network. Assuming \( M(i, j, t, \Delta t) \) is the amount of heat received by \( i \) from its neighbour \( j \) at time \( t \) in the period of \( \Delta t \) where \( f_j(t) - f_i(t) \) is the heat difference then

\[ M(i, j, t, \Delta t) = \alpha(f_j(t) - f_i(t))\Delta t \]

where \( \alpha \) is the thermal conductivity the heat diffusion coefficient.
Hence, this process can be formulated as:

$$\frac{f_i(t + \Delta t) - f_i(t)}{\Delta t} = \alpha \sum_{j: (v_j, v_i) \in E} (f_j(t) - f_i(t)) = \alpha H f(t) \quad (14)$$

Where

$$H_{ij} = \begin{cases} 
1 & (v_j, v_i) \in E \text{ or } (v_i, v_j) \in E \\
-d(v_i) & i = j \\
0 & \text{Otherwise} 
\end{cases} \quad (15)$$

Solving the differential equation, we have: $$f(t) = e^{\alpha t H} f(0)$$ in which $$e^{\alpha t H}$$ can be extended as:

$$e^{\alpha t H} = I + \alpha t H + \frac{\alpha^2 t^2}{2!} H^2 + \frac{\alpha^3 t^3}{3!} H^3 + \ldots, \quad (16)$$

An extension of the above model has also been defined for a directed graph and then a greedy algorithm has been proposed based on the above model to find the top most influential nodes in the network. This model has been evaluated on an Epinion dataset and the promise of their algorithm has been confirmed [20].

Kempe, Kleinberg and Tardos introduced two basic diffusion models; i.e., Linear Thresholds and Independent Cascade [52]. In these models, each node in the social network is considered to be either active (adopted/adopter to/of the innovation) or inactive.

**Linear Threshold Model**

The linear threshold model focuses on the setting that motivation for each node to be active monotonically increases as more of its neighbours become active [52–54]. In this model, a node $$v$$ is influenced by each of its neighbourhood nodes, $$w$$, with a weight $$b_{v,w}$$ such that $$\sum_{w \in N(v)} b_{v,w} \leq 1$$. In this model the propagation process
begins by choosing a threshold $\theta_v$ for each node representing the weight fraction of
neighbourhoods of $v$ that must be active to activate $v$. This model unfolds activation
of any node $v$ in a discrete manner, in step $t$ by determining whether the total weight
of its active neighbourhoods is at least $\theta_v$:

$$\sum_{w \in \text{ActiveN}(v)} b_{v,w} \geq \theta_v,$$  \hspace{1cm} (17)

Since this process is progressive, all of the active nodes in the step $t - 1$ remain
active. This model ignores the fact that, a node $v$ might be activated by one of its
neighbourhoods which is more trustable than the total weight of its active neighbour-
hoods. The invention of the Threshold Model dates back to 1978 by Granovetter and
Schelling [53].

**Independent Cascade Model**

The other model is called the Independent Cascade Model in which each individual
has a single chance to activate each of its inactive neighbourhoods [52,55,56]. The
probability of activating a node from one of its active neighbours is independent of the
set of active nodes which attempted to activate this node in the past. This process
can be described in a series of time steps in which each node $v$ tries to activate each of
its inactive neighbourhoods, $u \in \eta(v)$, with the probability $p(\text{activation}(u,v))$. This
model can be extended in such a way that the probability of activation of a node
from one of its neighbours depends on the others. This is called the General Cascade
Model. Formally, this latter model can be represented as $p(\text{activation}(u,v) \mid S)$ when
$S \subseteq N(u) \setminus v$ is the set of neighbours of $u$ who tried and failed to activate the node $u$.
This model is order independent, which means that the order of attempts from each
of its neighbours does not affect the probability of activation. A variant of this model
is defined as the Decreasing Cascade Model in which the probability of activation
of a node from each of its neighbours is non-decreasing in the process of activation. Formally, $p(activation(u, v) \mid S) \geq p(activation(u, v) \mid T)$ whenever $S \subseteq T$. After formulating the diffusion process, it is proved that the influence maximization problem is NP-hard. Finally, the researchers proved that the greedy algorithm guarantees the $(1 - 1/e)$ approximation since the influence function, $\sigma(A)$, is monotone and modular. Kleinberg proved monotonicity and sub-modularity of the influence function according to the aforementioned models of information diffusion [52, 56].

### 3.3 Recommender Systems

Recommender systems are engines providing services using information filtering techniques to propose information based on interests of a user [57, 58]. Typically, these systems have used a wide variety of techniques and parameters to choose information, products or services based on the users’ tastes. These techniques generally work based on mining of social interactions of the previous users, or based on the information provided implicitly or explicitly by the current user such as rating a product or explicitly defining his/her interests. There are several techniques used in this domain of research; such as Content based filtering, Collaborative Filtering and Social Recommender Systems. These are described in the next 3 sections.

#### 3.3.1 Content Based Filtering

Content based filtering is one of the most popular techniques for recommendation which works based on the information provided by the user (e.g., age, place of living, race, etc.). Some of this information can be implicitly inferred from a user according to the records saved by the user. For example, a record indicating a user did not enjoy his trip to Canada during the winter might indicate that the user does not enjoy winter activities or cold weather [59].
3.3.2 Collaborative Filtering

Another type of recommendation system is *Collaborative Filtering* in which a product or a service is recommended according to the similarity of tastes of different users or similarity of different products. For example, in the above example, a travel recommender system does not recommend Alaska to the people who did not enjoy their winter trip to Canada. Calculation of similarity between products or people is a key issue in this category of recommender systems. Simply put, in user-based collaborative filtering, the system tries to find people with similar tastes and recommend items highly ranked by the people which might be interesting to their peers. Finding people with similar tastes involves processing of their historical transactions (i.e., items viewed and ranked by them in their previous transactions) and calculating similarity between them using one of the methods described in Section 3.3.4 [57].

3.3.3 Social Recommender Systems

The third category of recommendation systems is *Social Recommender Systems* in which a product or a service is recommended by a friend to a person. This category of systems has a close relationship to viral marketing in which a product is promoted by a trusted person to a user in a market forming a social network. This category of information filtering is a motivating factor in considering the influence problem in this thesis. Similarity is also one of the issues in this research since the similarity of a product and the motivating application domain of interest for a user is a key issue in diffusion of information about a product. The next section is devoted to describing different methods for calculation of similarity between two entities [60].
3.3.4 Similarity Issue

The concept of similarity between two entities was defined in the previous chapter. In order to measure similarity between interests of a user on different topics and related topics of a posting in a social network, a model is required to measure this similarity. In the following paragraphs, several methods are described for measuring similarity between two entities.

Cosine Similarity

There are several existing techniques for measuring similarity between two entities represented by the vector of their features; such as Cosine Similarity, in which the cosine of the angle between two vectors represents the degree of similarity between those entities [61]. Formally:

\[
\text{sim}(e_a, e_b) = \cos(\theta) = \frac{e_a \cdot e_b}{|e_a| \cdot |e_b|},
\]

where \(e_a, e_b\) are the vectors of features for two entities.

Jaccard Similarity Coefficient

Jaccard Correlation is another alternative to measure similarity between two documents or entities regarding their attributes [62].

\[
J(A, B) = \frac{A \cap B}{A \cup B},
\]

Pearson product-moment correlation coefficient

Another method for measuring similarity is the Pearson product-moment correlation coefficient (PMCC). This method calculates the correlation (linear dependence) between two vectors. This method has been used in the simulation part of this research...
to calculate similarity between the vector of content and vector of interests of a user. This method is described in the following equation [61, 62]:

\[
\text{sim}(e_a, e_b) = \frac{\text{cov}(e_a, e_b)}{\sigma(e_a) \cdot \sigma(e_b)},
\]

(20)

**Multi-Tree Similarity**

All of these methods have pros and cons. Since in the above models, an entity is represented in a linear form (i.e., a vector of features), none of the aforementioned models is capable of capturing semantic relationship among the features of two entities. An illustration of the latter problem can be found in recommender systems which find people with similar tastes according to their previous transactions. An example of this problem is demonstrated in the following example in which similarity of tastes for two people are estimated based on their previous transactions [63]:

- T1 (Clothes, Boxspring, Mp3Player, Mattress, LCD TV)
- T2 (Dress, Bed, Mattress, iPod Touch, LED TV)

Using the VSM-based method for computing similarity between the above transactions, these transactions are no longer similar at all. However, intuitively we have a feeling that LED TV and LCD TV are related to each other since both are subclasses of TV. This observation is also true when comparing iPod Touch and Mp3 Player and for the relationship between Clothes and Dress.

There are many papers, such as [64, 65], that describe how to use an ontology in order to enhance search results. Some of them consider relations among entities for measuring similarity between two objects. In a research related to this thesis, we proposed a novel method for measuring similarity using a non-linear multi-tree structure for representing two entities instead of VSM. A multi-tree is defined as a
directed graph in which each node may have more than one parent. The architecture of this system is shown in the following figure [63]:

![Diagram of Multi-Trees process]

**Figure 6:** Calculating similarity using Multi-tree model

In this method, two multi-trees representing two entities are combined and the similarity is calculated based on not only the commonality of leaves but also the commonality of their parents. This method recursively assigns a weight to each node according to the similarity of the sub-tree rooted on the node. This weight is propagated to the root of the multi-tree. The root of the multi-tree indicates the similarity of the two entities. The method has been evaluated for the application of measuring similarity between two words using Wikipedia Categories as a taxonomy on the WordSim-353 dataset [63].

Every method has its advantages and disadvantages in terms of precision and complexity. Later, in Chapter 6, we describe our simulation and, in the simulation, *Pearson product-moment correlation coefficient* has been used due to its low computational complexity, simplicity of implementation and efficiency in terms of precision. Therefore, as future work this model can be a good alternative as a replacement for VSM when VSM is inadequate.
3.3.5 Social Recommender Systems and Viral Marketing

There are several works concerning Social Recommender Systems which are among the most recent generation of recommender systems [60]. In fact, recommender systems generally rely on social interaction of users and websites. However, social recommender systems explicitly consider social ties between people for marketing new products. There are several models found in the trust literature for peer to peer recommendation. These systems rely on the fact that people like to get recommendations from the people they trust rather than automated software. The following issues can be summarized for these systems [66]:

- **Information Filtering**: Delivery of right contents to the right person according to his/her interests.

- **Determining the customers’ needs**: The customers’ needs are important in delivery of appropriate information to the right person.

- **Trust and reputation issue among users**: The people do not usually trust every comment or every rating for a product or a service. Regarding this issue, a trusted person should recommend a product to a consumer. The reputation of a user in a society is also important.

- **Promoting a product to valuable customers**: Some marketers are interested in promoting their new products by offering some form of discount to their customers. With this issue, we are concerned with finding the most valuable customers who are more trustable and have the most influence on their friends.

- **Cold Start problem**: Lack of sufficient information for drawing inference before gathering information about users’ interests [5] is a problematic issue in these systems.
The process of recommending a new product in this category of recommender system is sometimes called *viral marketing* [60]. In *viral marketing* customers help marketers to promote a product or a service. This method is often a more cost effective type of advertising and, in some cases, more effective in absorbing new customers and making people adopt a new product because people are more affected by their friends or the people they trust. This method is sometimes called "word-of-mouth" advertising. In certain cases, imitation is a determining factor in promoting a product [9]. On the other hand, some people like to perceive themselves as unique and get the newest products to be distinguished from their society. In marketing these two categories are target people for marketing new products. However, we are looking for an effective marketing plan to reduce computational and marketing costs with an optimal marketing value [9]. It is this domain problem that motivates the research described in this thesis.

A key question in this category of marketing is finding the best set of people for optimizing market value. Richardson and Domingos introduced a probabilistic model considering the discount offered to some customer and parameters such as imitation and self reliant factors [67]. This model determines the probability of buying product $Y$ by the user $i$ given set of his/her neighbourhoods (or friends) influencing him/her and marketing action set $M$ taken by the marketers:

$$P(X_i \mid X \setminus X_i, Y, M) = P(X_i \mid N_i, Y, M)$$

$$= \beta_i P_0(X_i \mid Y, M) + (1 - \beta_i) P_n(X_i \mid N_i, Y, M), \quad (21)$$

where $X_i$ is a random variable which takes the value 1 if the customer $i$ buys the product $Y$ and has the value 0 otherwise. $N_i$ is the set of neighbours of $X_i$ influencing him, and $0 \leq \beta_i \leq 1$ is the factor describing how self reliant $X_i$ is [67].
The marketer’s goal is to find the marketing plan, $M$, such that maximum profit is achieved. $M$ is a boolean vector in which each element represents the marketing action for a specific product. Furthermore, the model is extended using expected lift in profit (ELP) function and customer’s network value is defined as a difference between his total and intrinsic value in the extended model. A customer with a high network value is the one who, when marketed to, influences many others to purchase the product. The above model was applied to an Epinion dataset by Domingos and Richardson and their first results show a 10-18 percent improvement in profit gained by Viral Marketing in comparison with No-Marketing or Direct Marketing strategies [9,67].

On the other hand, we have social recommender systems which are trying to decrease the information overload over social media and deliver the right contents to the right person. In these systems people benefit from other users’ comments and ratings on specific products. In other words, people are situated in a referral network in which a service or a product is recommended by his/her friends in a social network [60].

3.4 Trust in Social Networks

Trust is one of the determining factors in propagation of information in social networks. As discussed in Chapter 2, trust is defined as the impression of an individual about its friends. This impression affects the decision of re-sharing of a posting by an individual in a network due to belief of that individual about trueness of the information. This section discusses some of the previous literature in the domain of trust and reputation models. These models can be applied to the historical data about the past interactions between two users. Since trust is time dependent, a model is required to update the value of trust between two individuals according to the history of their interactions. In this thesis, diffusion models will be defined in Chapter 4 in
which the probability of re-sharing of a posting by an individual is considered as a function of trust, similarity, and laziness of the target agent. Trust is a factor of belief of individuals about the trueness of a posting in a social network. As an example, a posting might be spam or not-spam in a social network. Each of the following models has advantages and disadvantages in terms of robustness to different attacks. The study of these attacks is beyond the scope of this thesis.

### 3.4.1 Yu and Singh Trust Model

Yu and Singh modelled trust by $T_{i,j}(t) \in [-1, 1], T_{i,j}(0) = 0$ indicating how agent $j$ is trusted by the agent $i$ [68]. The degree of the trustworthiness of agent $j$ by agent $i$ changes as a result of their interaction sequences, or transactions. Cooperation and defection in the transactions between agent $i,j$ are represented by positive evidence, $\alpha$, that $0 < \alpha < 1$ and negative evidence, $\beta$, that $-1 < \beta < 0$ respectively. The trust value between these agents changes according to the following rules in table 2:

<table>
<thead>
<tr>
<th>Condition 1</th>
<th>Condition 2</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{i,j}(t) &gt; 0$</td>
<td>Cooperation</td>
<td>$T_{i,j}(t + 1) := T_{i,j}(t) + \alpha(1 - T_{i,j}(t))$</td>
</tr>
<tr>
<td>$T_{i,j}(t) &lt; 0$</td>
<td>Cooperation</td>
<td>$T_{i,j}(t + 1) := (T_{i,j}(t) + \alpha)/(1 - \min(</td>
</tr>
<tr>
<td>$T_{i,j}(t) &lt; 0$</td>
<td>Defection</td>
<td>$T_{i,j}(t + 1) := T_{i,j}(t) + \beta(1 + T_{i,j}(t))$</td>
</tr>
<tr>
<td>$T_{i,j}(t) &gt; 0$</td>
<td>Defection</td>
<td>$T_{i,j}(t + 1) := (T_{i,j}(t) + \beta)/(1 - \min(</td>
</tr>
</tbody>
</table>

**Table 2:** Yu and Singh Trust model update rules

Yu and Singh discuss how trust is propagated across a network of agents by means of the concept of reputation propagation. This fact was modelled according to the incorporation of testimonies from different witnesses. Then, the phenomenon of gossip in such networks is discussed and finally, their experimental results demonstrate the usefulness of their model according to the effectiveness of selection of rewards and penalties by simulating agents [68].
3.4.2 Regret Trust Model

The Regret model considers $T_{i,j}(t)$ as a weighted mean of the impressions’ rating factors in which the model gives more weight to more recent impressions [69]. Assuming $r_{i,j}(t)$ as an impression of agent $i$ about agent $j$ at time $t$, where $0 < r_{i,j}(t) < 1$, $R_{i,j}$ as the set of $r_{i,j}(t)$s for all possible $t$. The subjective reputation of agent $j$ from the point of view of agent $i$ at time $t$ is measured by the following equation:

$$T_{i,j}(t) = \sum_{\forall r_{i,j}(t_k) \in R_{i,j}} \rho(t, t_k)r_{i,j}(t_k)$$ (22)

where $t$ is the current time, $t_k$ is the time that $r_{i,j}(t_k)$ is recorded and $\rho(t, t_k)$ is called recency factor calculated by the following equations given $f(t_k, t) = t_k/t$:

$$\rho(t, t_k) = \frac{f(t_k, t)}{\sum_{\forall r_{i,j}(t_k) \in R_{i,j}} f(t_k, t)}.$$ (23)

3.4.3 FIRE Trust Model

Fire has the same trust model as Regret except that the recency function is slightly different from Regret. If we replace the recency function of the Regret trust model by the following recency function, the FIRE model is achieved. In this function, $\lambda$ is the scaling time parameter [24].

$$f(t_k, t) = e^{-\frac{t - t_k}{\lambda}},$$ (24)

3.4.4 TRAVOS Trust Model

TRAVOS uses a probabilistic trust model which has recently gained considerable popularity [70]. The Beta Reputation System and TRAVOS both use a probabilistic model and beta distribution as the confidence in the trust value. The probabilistic
trust systems are modelled based on the premise that past interactions are an indicator of the observation of defection or cooperation in future. In TRAVOS, $B_{i,j}$ is the probability that obligations of agent $j$ will be satisfied by agent $i$. $O_{i,j}(t)$ is the outcome of the contract between $i$ and $j$ at time $t$. Therefore,

$$O_{i,j}(t) = \begin{cases} 1 & \text{if contract is fulfilled by } j \\ 0 & \text{Otherwise} \end{cases}$$

$R_{i,j}(t) = (m_{i,j}(t), n_{i,j}(t))$ is the set of historical interactions between two agents at time $t$ where $m_{i,j}(t)$ is a successful set of interactions and $n_{i,j}(t)$ is an unsuccessful set of interactions between two agents.

$B_{i,j} = P(O_{i,j} = 1)$, where $B_{i,j} \in [0,1]$ and the trust value between $i$ and $j$ is determined by $T_{i,j}(t)$:

$$T_{i,j}(t) = E(B_{i,j} \mid O_{i,j}(t) = 1),$$

As the standard equation for expected value for the beta distribution, we have: $E(B \mid \alpha, \beta) = \frac{\alpha}{\alpha + \beta}$, we have:

$$T_{i,j}(t) = E(B_{i,j} \mid O_{i,j}(t) = 1) = \frac{\alpha}{\alpha + \beta},$$

where $\alpha = m_{i,j}(t) + 1$ and $\alpha = n_{i,j}(t) + 1$ denoting cooperation and defection respectively [70].

### 3.4.5 AER Trust Model

Researchers have demonstrated the possibility of exploitation in artificial societies of trust and reputation models. Recent work has examined the attack resistance of different trust models [71]. These attacks have been analyzed on different trust
models in the domain of e-commerce marketplaces. In the con-man attack scenario, a malicious agent can gain a high trust value by using cyclical behaviour and then misuses the gained trust by committing a bad transaction. All of the aforementioned models can be attacked using con-man attack scenario.

The AER trust model is an extension of the Yu and Singh model where $\alpha, \beta$ are themselves parameters of defections and cooperation in the past transactions. A new parameter $\gamma_d$ is added for the negative evidence weighting when an agent perceives defection. In AER, we have [71]:

\[
\alpha(i) = \alpha(i-1) \times (1 - |\beta(i-1)|) \\
\beta(i) = \alpha(i-1) - \gamma_d \times (1 + |\beta(i-1)|)
\] (28)

where $\gamma_d$ is a discounting factor and is in the range of $[0,1]$. $\alpha$ and $\beta$ are being updated whenever a defection occurs. This adaptation prevents the model from exploitation by the con-man attack [71].

### 3.5 Evolution in the Networks and Referral Networks

Evolution in social media and social networks’ structure is a natural phenomenon due to the dynamics of social media. The structure of a network is always changing during the activities in these networks [72,73]. People always find new friends in their real life and cut ties to others according to their interests and interactions with those friends. The weights of these relations are also dynamic as a consequence of the changes in the trust value between pairs of users in these friendships. This phenomenon is important is this thesis because propagation of information not only depends on the behaviours
of individuals but also depends on the structure of the network. As the structure of a network changes with time, this fact needs to be taken into consideration when we are studying propagation of information in social networks. This section is important for the further study of the effect of different structures in propagation of information in social networks which is beyond the scope of this thesis and is defined as future work.

The evolution in the structure of social networks can occur through the use of referral in these networks. The concept of a referral network dates back to the use of a referral system in multi agent systems in which an agent introduces another agent as an expert in a specific domain according to a query received from its neighbour. Referral systems help dissemination of concept knowledge through referrals to other sources of knowledge and consequently the concept knowledge is propagated through the network [72, 73].

The scenario of referral is as follow: an agent queries some of its neighbours and the queried agents may provide the requested service or may give referrals to their neighbours according to referral policies [72]. The agent which asked the query may accept the answer or pursue the referral and send the query to a referred agent as an expert. There are certain policies as determining factor of the quality of the system. This system is applicable in real life. Consider the scenario of querying search engines several times. As an example, once we pose a query to Google search for a scientific paper, Google may return some results from websites such as SpringerLink, ACM, or IEEE databases. On the other hand, Google corrects the spelling and may suggest new keywords which are more relevant to the desired results. In the consequent steps, we may browse the referred databases with new keywords for finding more relevant results.

There are several evaluation metrics specified in referral networks. We briefly describe them in below [72, 73]:
• *Similarity* discussed above is a determining factor in referral policy. An agent may refer an agent to its neighbours according to the similarity of vector of interests for the querying agent and referred agents. The similarity has been discussed above in Section 3.3.4

• *Capability* is the degree of certainty of the replying agent on its expertise in the topics of the query. This metric is calculated according to the cosine similarity between the vector of the query and that of the agent’s interests.

• *Effectiveness* measures how easily a provider can be found. This metric is calculated according to the number of agents that need to be asked to reach an agent whose vector of interests is more similar to the corresponding agent.

• *Efficiency* is measured according to the accuracy of the responses obtained from an agent.

• *Authoritativeness* is the degree of popularity calculated according to the PageRank of an agent. This metric corresponds to the greater number of well-connected agent which are connected to the corresponding agent.

• *Health* is a metric demonstrating how well an agent has learned according to the concept knowledge received.

Referral systems are applicable in viral marketing for marketing services or a product [72, 73]. A referral system presents just one of the mechanisms for changing network structures. A referral system changes the structure of a network by adding links between an agent and its friend of a friend and by removing some links according to the unusefulness of a neighbour regarding the lack of quality of service or sociability. Another determining factor in changing the network structure is expertise and popularity. According to experience, agents tend to make direct friendships with other nodes with high values of reputation, popularity and expertise. This fact causes
popular nodes in open networks such as Twitter to be followed by more and more people across the network.

Referral networks are included here as they provide mechanisms that could be incorporated into socially-based recommender systems. Furthermore, they incorporate considerations of similarity, trust and reputation and information diffusion which are considered central to a model of influence which is the topic of this thesis.

3.6 Summary

This chapter provided an analysis of related works pertinent to the determination of influential nodes in a social network. This task involves social network analysis. Two categories of social network analysis were described: *structural analysis* and *behavioural analysis*. As propagation of information in a social network takes effect from both categories of analysis, both approaches were analyzed. Several metrics were defined in the area of structural analysis such as Closeness, Betweenness, etc. On analyzing related work in behavioural analysis it was noted that diffusion of information in the network is required for modelling. Consequently, this chapter provides information about several diffusion models studied by other researchers.

As motivation for this research, recommender systems were described and different methods for recommending products and services were analyzed. According to the pros and cons for each of the previous methods, a new generation of recommender systems are highlighted; namely, social recommender systems. The results included in the chapters which follow provide key answers for implementation of these systems. Therefore, issues and difficulties in recommender systems are discussed in this chapter. Similarity and trust are other related research concepts which have a significant effect on the models proposed in this thesis. As an example, trust is the belief of an individual in the trueness of a piece of information. This belief affects the behaviour
of an agent in the propagation of a posting in a network. On the other hand, the degree of nearness of interests of an individual in related topics of a posting is important.

Finally, related work in the evolution of behaviours of individuals and the structure of a network is described. This part is described as potential work for the future. The next chapter is devoted to the formulation of the problem of influence measurement and maximization in social networks.
Chapter 4

Formulation of Problem

4.1 Introduction

This chapter is devoted to the formulation of the problem of measuring of influence in a social network. This chapter formalizes the measuring of influence according to the propagation of information in the network. We can categorize the problem of maximizing influence into two different approaches: (i) measuring the influence of a node in a network in that the focus is on rating nodes in the graph based on their importance according to their capability of infecting a larger number of nodes; (ii) finding a k-element set of nodes such that this set covers the maximal number of nodes in the graph by infecting other nodes. In the latter, we are interested in choosing a set of nodes such that their combination covers the maximum number of nodes in a graph since each node in the covering set corresponds to a set of nodes being infected by the nodes in the covering set as starting point. The problem solved in this thesis is the first problem described above.

In this chapter we use the observation that the transmission of information from one node to another one is a random process. This fact is formally modelled with a probabilistic model described in the following sections.
4.2 Definitions and Modelling of a Social Network

The formally defined social network included in this section is limited to re-sharing of information by the individuals or users of the social network. On the other hand, this specific definition of behaviours can be generalized to more general networks such as diffusion of diseases among the network of humans or diffusion of advertisements in the network of an economic social network. We limit our attention to the spread of information in the social network of simple friendships such as Facebook and the limited behaviours such as posting creation and re-sharing of postings in this network.

Formally, a social network, $SN = (G, P)$ is defined by its structure represented by a graph $G = (V, E)$ and set of contents or postings $P$ generated by the individuals in the network.

The graph $G = (V, E)$ includes a set of vertices $V$ representing the individuals which may have some dynamic behaviours and set of edges $E$ representing the relationships (or friendships) among them.

- Each vertex $v_i \in V = \{v_1, ..., v_n\}$ represents an individual in the network. We say the $i^{th}$ individual is represented by a tuple $(B_i, Prop_i, DM_i)$ demonstrating its behaviours $B_i$, properties $Prop_i$ and decision making strategy $DM_i$ which is the strategy of the $i^{th}$ individual for re-sharing of a posting in a social network.

  - $B_i = \{PC, PP, Idle\}$ is the set of behaviours including post creation $PC$, post propagation $PP$ and $Idle$.

  - $Prop_i = \{W_i, VI_i, PCR_i, PPR_i, T_i\}$ is the set of properties of the vertex $v_i$ comprising:

    * $W_i = \{p_{i1}, ..., p_{il}\}$ as a wall which is a set of postings created by the corresponding agent or re-shared from its neighbours,
* $VI_i = < I_{i1}, ..., I_{im} >$ as a vector of interest, in which $\forall I_{ik} \in VI : I_{ik} \in [0, 1]$ represents the levels of interest of the individual $i$ on topic $k$, where increasing value of each $I_{ik}$ determines an increasing level of interest on the topic $k$.

* $PCR_i$ as the post creation rate refers to how likely an individual creates a posting in the network. This concept is also referred as the laziness of the $i^{th}$ individual in creating content in the social network. This concept considers the differences in intention of people in a society in terms of being active in creating new posts [74]. This property generally varies with time; however, in this research we assume a snapshot of a network. Therefore this property is constant for each node in this thesis.

* $PPR_i$ is the post propagation rate referring to the likeliness that an individual propagates a posting from one of its friends. This concept is also referred as the laziness of the $i^{th}$ individual in infection to a posting from one of its neighbourhoods [74]. This property generally varies with time; however in this research this property is considered as a constant as, a snapshot of the network is considered in computing influence.

* Given $m$ different topics, for each topic $k$, $(1 < k < m)$, $T^k_i$ is the trust set associated with the individual represented by the vertex $v_i$ regarding the impression of the corresponding individual about his/her friends on a specific topic. We have $T^k_i = \{T^k_{i1}, ..., T^k_{in}\}$ in that each tuple $T^k_{ij} \in T^k_i : T^k_{ij} \in [-1, 1]$ represents the impression (trust value) that the $i^{th}$ individual has about $i^{th}$ individual on the $k^{th}$ topic.

- Each edge $e_i = < a, b >$ is an ordered set that $e_i \in E$ represents a directed
relation from vertex $a$ to $b$. In some models, $e_i = \{a, b\}$ represents a tie between $a$ and $b$ which is called an undirected edge that ignores the direction of the relationships. In some social networks such as Twitter the nodes are connected to each other by directed edges. In these networks, user $a$ is said to be following $b$ when in the graph of their network there exists an edge $e = <a, b>$ connecting user $a$ to $b$. Formally:

$$\text{follows}(a, b) \iff \exists e = <a, b> \in E$$

(29)

$\mathcal{P} = \{p_1, \ldots, p_N\}$ is the set of postings, where $(1 < k < N) : p_k = (v_i, VR_k, S_k, C_k, R_k)$ is a tuple of a posting which includes:

- $v_i$ is the corresponding vertex representing the $i^{th}$ individual who posted this posting.
- $VR_k = <t_{k1}, \ldots, t_{km}>$ as a vector of relatedness on topics where each $t_{kl} \in [0, 1]$ represents the relatedness of the posting $k$ to the topic $l$.
- $C_k = \{c_{k1}, \ldots, c_{kn}\}$ is a set of comments about the posting $p_k$. Every $c_{kj} \in C_k$ is a string of characters representing the comment of the $j^{th}$ individual $(1 < j < n)$ about the posting $p_k$.
- $S_k$ is a string representing the content of the posting $p_k$.
- $R_k = \{r_{k1}, \ldots, r_{kn}\}$ is a set of ratings from other agents on a posting $p_k$. A rating $r_{kj} \in R_k : r_{kj} \in [-1, 1]$ represents the degree of interest of the $j^{th}$ individual in the posting $p_k$. 

4.3 Modelling Information Diffusion

In an independent cascade model, when a node shares a posting $p_k \in \mathcal{P}$, the node is called infected. On the other hand, once a node $v_i$ becomes infected, all of its neighbours $\forall v_j \in \eta(v_i)$ get a single chance to be infected from $v_i$. In this case, we say the node $v_i$ is infectious or active for infecting his neighbours. After giving a single chance of sharing the posting $p_k$ with each neighbour, the node is no longer active for infecting its neighbourhoods but it stays infected.

**Definition** A node $v_i$ is called infected according to the posting $p_k \in \mathcal{P}$ if the posting $p_k$ belongs to the wall of the node $v_i$ but it is called active if it is capable of infecting its neighbourhoods. Formally, the function $\lambda^p_k(v_i)$ is equal to 1 if the node $v_i$ is infected according to the post $p_k$ otherwise, it is not infected.

\[
\lambda^p_k(v_i) = \begin{cases} 
1 & p_k \in \mathcal{W}_i \\
0 & \text{Otherwise} 
\end{cases}
\] (30)

where, $\mathcal{W}_i$ is the wall of the node $v_i$.

**Definition** $\zeta^p_k(v_i, v_j)$ is a function mapping each pair of nodes $(v_i, v_j)$ to 1 if the node $v_i$ is infectious to his neighbour $v_j$, otherwise its value is equal to 0. An infectious node i.e., $v_i$ is also called active for activating $v_j$.

**Definition** $\omega^p_k(v_i, v_j)$ is the probability that the $j^{th}$ individual gets infected from the $i^{th}$ individual by re-sharing the posting $p_k$ from the $i^{th}$ individual. This probability is a function of other factors which will be defined in the following sections.
4.3.1 Deterministic Independent Cascade Model

In this model, every node (e.g., \( v_j \)) has a property as a threshold probability of infection \( \theta_j \) which is the characteristic of the \( j^{th} \) individual. In this model, a node \( v_i \) is capable of activating \( v_j \) by the posting \( p_k \), if the node \( v_i \) is infectious for \( v_j \) and the probability of transmission of \( p_k \) from \( v_i \) to \( v_j \) is greater than or equal to a threshold \( \theta_j \). The threshold \( \theta_j \) depends on the characteristics of the \( j^{th} \) individual. For instance, an individual might be an optimist, pessimist or conservative in terms of sharing a piece of information and the threshold depends on these qualities of the individual. The following equation defines the activation function for a specific node in the network according to this model:

\[
\Lambda_{pk}(v_i, v_j) = \begin{cases} 
1 & \zeta_{pk}(v_i, v_j) = 1 \text{ and } \omega_{pk}(v_i, v_j) > \theta_j \\
0 & \text{Otherwise}
\end{cases}
\] (31)

4.3.2 Non-deterministic Independent Cascade Model

In this model, a node \( v_i \) is capable of activating \( v_j \) by the posting \( p_k \), if the node \( v_i \) is infectious for \( v_j \) and the value of a random variable \( d \) is less than or equal to the probability of transmission of the posting \( p_k \) from \( v_i \) to \( v_j \). Formally,

\[
\Lambda_{pk}(v_i, v_j) = \begin{cases} 
1 & \zeta_{pk}(v_i, v_j) = 1 \text{ and } d < \omega_{pk}(v_i, v_j) \\
0 & \text{Otherwise}
\end{cases}
\] (32)

4.3.3 Probability of Transmission

The probability that a piece of information \( p_k \in \mathcal{P} \) is transferred from the node \( v_i \) to the node \( v_j \) depends on several factors. In this thesis, the following factors are considered as determining parameters of this process:
Similiarity between the corresponding topics of a piece of information and the interests of the node to which the information is transmitted. The function $0 \leq \Psi(v_j, p_k) \leq 1$ returns the degree of interest of the node $v_j$ in the posting $p_k$. In other words, this function is defined as the similarity between the vector of interests of the $j^{th}$ individual and the vector of relatedness of the posting $p_k$. According to the social network modelled in Section 4.2:

$$\Psi(v_j, p_k) = \text{sim}(v_j.VI, p_k.VR),$$

where $VI$ is the vector of interests for the $j^{th}$ individual and $VR$ is the vector of relatedness to topics for the posting $p_k$.

- **Trust** value between the users $v_j$ and $v_i$ based on the related topics of the posting $p_k$. The function $-1 \leq \tau^{p_k}(v_j, v_i) \leq 1$ returns the level of trust between users $v_j$ and $v_i$ according to the posting $p_k$. This function returns the corresponding value from the set of trust values between two individuals in the network described in Section 4.2. This parameter is a time varying parameter; however in this thesis, we consider a snapshot of the network [75].

- **Laziness** of the user $v_j$ to which the posting $p_k$ is transmitted. This value is given by the function $0 \leq \Gamma(v_j) \leq 1$. This parameter is actually one of the properties of the node $j$. This parameter refers to the post propagation ratio $PPR_j$ described in Section 4.2 [75].

Now, we define the probability of transmission of the posting $p_k \in \mathcal{P}$ from user $v_i$ to $v_j$. The function $\omega^{p_k}(v_i, v_j)$ returns the aforementioned probability. This parameter can be represented as a weight of an edge between two vertices in the graph representing a social network which is shown in Figure 7. The omega function in the equation below is the capability of $v_i$ in infecting $v_j$:
\[ \omega^{pk}(v_i, v_j) = Pr(\Lambda^{pk}(v_i, v_j) \mid \zeta^{pk}(v_i, v_j)) = f(\Psi(v_j, p_k), \tau^{pk}(v_j, v_i), \Gamma(v_j)), \] (34)

**Figure 7:** Transmission probability of infection of a vertex \( v_j \) from its neighbour \( v_i \)

### 4.4 Activation Probability of a Node in a Network

It is stated in Chapter 3 that, in the independent cascade model, whenever a node becomes active, each of its neighbourhoods has a single chance to be activated in a random process with the probability \( \omega(v_i, v_j) \) independent of the history of other neighbours trials in activating \( v_j \). The independence means that if the node \( v_j \) has multiple neighbours, their attempts to activate \( v_j \) can happen in an arbitrary order and independently of each other’s attempt.

As shown in Figure 8, although the probability of activation of a node from one of its neighbours in a network is independent of trials from other neighbours, the probability that \( v_j \) becomes activated by \( v_i \) in a network depends on the probability of activation of \( v_i \) times probability of transmission from \( v_i \) to \( v_j \). In other words:

\[ Pr(\Lambda^{pk}(v_i, v_j) = 1) = Pr(\zeta^{pk}(v_i, v_j) = 1) \times \omega^{pk}(v_i, v_j), \] (35)

**Definition** The path \( path^{pk}(v_s, v_j) \) is called an active path iff \( \forall < v_n, v_m > \in path^{pk}(v_s, v_j) : \Lambda^{pk}(v_n, v_m) = 1 \) then the edge \( < v_n, v_m > \) is called an active edge.
If we have a starting node \( v_s \) in a network which is currently active, the probability that any arbitrary node \( v_j \) is activated starting from \( v_s \) can be measured by finding all of the paths from \( v_s \) to \( v_j \). The following equation returns the probability of activation of \( v_j \) given \( v_s \) is already activated:

\[
Pr(\lambda^p(v_j) = 1 | \lambda^p(v_s) = 1) = 1 - \prod_{\forall Pa \in \text{path}^p(v_s,v_j)} (1 - \prod_{\forall <v_n,v_m> \in Pa} \omega^p(v_n,v_m)), \quad (36)
\]

where \( \text{path}^p(v_s,v_j) \) returns all of the paths starting from node \( v_s \) ending at \( v_j \). In the above equation \( \prod_{\forall <v_n,v_m> \in Pa} \omega^p(v_n,v_m) \) is the probability of activation of \( v_j \) from \( v_s \) through the path \( Pa \) where \( <v_n,v_m> \) denotes the directed edge from \( v_n \) to \( v_m \) in the corresponding path.

This problem can be mapped into the network reliability problem in which we try to measure the probability that a packet of data is delivered to a specific node in a network [76]. Most of these problems are NP-complete and can be estimated by a heuristic search algorithm. However, definition of these problems in a directed acyclic
graphs makes them solvable with a polynomial algorithm [76].

### 4.5 Symmetric Model Vs. Asymmetric Model

A diffusion model is called symmetric if the transmission probability of $p_k$ from $v_i$ to $v_j$ is equal to that of $v_j$ to $v_i$. This model can be represented by an undirected graph. Formally, in a symmetric model:

$$
\omega^{p_k}(v_i, v_j) = \omega^{p_k}(v_j, v_i),
$$

However, in the asymmetric model, the ability of activation of $v_j$ by $v_i$ does not indicate the ability of activation in the other direction.

The symmetric diffusion model is used for modelling the spread of viruses or information in the network in which the existence of a connection between two nodes is sufficient for transmission of a disease between two nodes. However, in many real world systems, this model is not particularly useful since trust between two nodes is not symmetric. As an example, infection by a posting in a social network is not equal to the chance of infection in the other direction. This observation makes an asymmetric model more useful in modelling real world systems.

### 4.6 Influence Maximization Problems

In this section, two problems are defined: the problem of maximizing influence in a social network which focuses on covering a larger number of nodes by selecting appropriate nodes as starting points for spreading a piece of information through a network. Key questions posed here are: how to measure influence and what are the criteria for determining the influence of a node in a social network.
4.6.1 Measuring Influence in a Social Network

**Definition** The infection function $\delta(v_s)$ maps every node $v_s \in V$ from the set of vertices in the graph to the expected set of nodes infected in the graph starting from $v_s$. The function $\eta$ returns the immediate neighbourhoods infected from $v_s$.

$$\delta(v_s) = \eta(v_s) \cup \left( \bigcup_{\forall v_l \in \eta(v_s)} \delta(v_l) \right),$$

(38)

The problem of measuring influence of each node in a social network is equivalent to rating nodes in the graph according to the ability of each node to infect other nodes in the network. In other words, influence is measured according to the ability in propagation of information. Formally, a node $v_i$ is the most influential node in the network iff:

$$\forall v_j \in V : |\delta(v_i)| \geq |\delta(v_j)|.$$  

(39)

The above equation models influence as a recursive concept that considers influence of a node as a function of influence of its neighbourhoods. This definition of influence has already been done by us in the concept of influence rank in which the most influential node is the one whose friends are also influential. In [37], we have modelled influence as ratio of feedbacks that all of postings of each individual gets from his/her friends in a social network.

4.6.2 Influence Maximization Problem

**Definition** The influence function $\sigma(A)$ maps each subset $A \subseteq V$ to the expected set of infected nodes when the set $A$ is targeted for initial activation. In this definition, $V$ is the set of vertices in the graph $G = (V, E)$.

$$\sigma(A) = \bigcup_{\forall v_i \in A} \delta(v_i).$$

(40)
The problem of finding a set of $k$ influential nodes is to find a $k$-element subset $A \subseteq V$ such that for all $k$-element subsets $A'$ the cardinality of the expected set of infected nodes starting from initial set $A'$ will be less than that of $A$ [77]. Formally,

$$\forall A' \subseteq V : |\sigma(A)| \geq |\sigma(A')| \text{ and } |A'| = k.$$ (41)

### 4.7 Measuring Influence in the Deterministic Cascade Model

In the deterministic model, the influence of a node can be easily determined according to the number of nodes which are infected from a starting node. We call this set the influence set. This problem can be easily solved by starting from a node and running a breadth first search on the graph. The complexity of breadth first search (BFS) [33] given in Algorithm 1 is $O(|V| + |E|)$.

**Algorithm 1** BFS for measuring the influence of a node in a network

```plaintext
Function VertexSet BFS(\(v_s\), Posting \(p\))
    queue Q;
    VertexSet \(\delta\);
    Q.enqueue(\(v_s\));
    \(\delta = \delta \cup \{v_s\}\)
    while Q is not empty do
        v=Q.dequeue();
        for all \(v' \in \eta(v)\) do
            if \(\omega^p(v,v') \geq \theta_{v'}\) and !\(\delta\).contains(\(v'\)) then
                Q.enqueue(\(v'\));
                \(\delta = \delta \cup \{v'\}\)
            end if
        end for
    end while
    return \(\delta\);
```

Now, if we calculate \(\delta\) for all of the nodes in the graph, the cardinality of the
corresponding set represents the influence of a node. Please note the influence sets for all nodes in the influence set are the same as a consequence of the symmetric nature of the model. Therefore, we can infer that the symmetric model partitions the network into several disjoint sets each of which has a cardinality representing the influence of nodes included in the set. This observation is central to the proof of theorem 4.8.2 which in turn supports the proof of theorem 4.8.4.

4.8 Influence Maximization in the Symmetric Deterministic Cascade Model

In this model, any node \(a_i \in A_i\) represents the subset/partition \(A_i \subseteq V\) where, by activating any node in \(A_i\), all of the other nodes \(v_j\) in the corresponding set are activated. The reason is that there is an active path between all of the pairs of the nodes in \(A_i\). Formally, \(\forall v_i, v_j \in A_i \exists \text{ActivePath}(v_i, v_j) \in \text{Path}(v_i, v_j)\).

**Definition** A node \(v_j\) is *reachable* from node \(v_i\) if there exists an active path between two nodes. The reachability of a node is interpreted as activation of a series of nodes starting from one node and ending at another node in the network.

**Lemma 4.8.1** The reachability relation is a transitive relation.

**Theorem 4.8.2** Assuming the symmetric model, the reachability relation partitions nodes in a network into several disjoint sets of nodes (i.e., connected components of a graph).

**Proof** Proof by Contradiction: assume \(A \neq B\), \(A\) and \(B\) as overlapping covering subsets of \(V\), therefore, \(A \cap B \neq \emptyset\). Hence, \(\exists v_k \in A, v_k \in B\). According to the transitivity property of reachability, \(v_k\) is reachable to/by all of nodes in \(A\), and all of the nodes in \(B\) are also reachable by/to \(v_k\). Therefore, all of the elements of \(A\) are
reachable by all of the elements in $B$, and all of the elements in $B$ are also reachable by all of the elements in $A$. Hence, $A = B$ which is contradictory to our assumption.

**Theorem 4.8.3** There exists a polynomial algorithm to find the $k$-element cover set of the graph in the symmetric deterministic cascade model.

**Proof** Theorem 4.8.2 indicates that reachability relation partitions a graph $\mathcal{G}$ into disjoint sets or, in other words, into sets of connected components $S = \{A_1, A_2, ..., A_{CC}\}$, where $\mathcal{V} = \bigcup_{A_i \in S} A_i$, and $\bigcap_{A_i \in S} A_i = \emptyset$. Therefore, $\forall A_k \in S, \forall v_i, v_j \in A_k \Rightarrow \text{Reachable}(v_i, v_j)$. Each connected component represents a set of vertices $A_i$ in which all of the elements are reachable to/from one another. According to Hopcroft and Tarjan [78] the breadth first search finds the connected components in a graph in linear time which is demonstrated in Algorithm 2. A result of this algorithm on a graph can be observed in Figure 10. Since every set $A_i$ is represented by one of its elements to cover the whole set a $k$-element cover set of a given graph can be constructed by randomly choosing a vertex $v_i$ from a set $A_i$, where $A_i$ is chosen from $S$ regarding its cardinality. Please note that at most $k$ connected components are sampled in this problem. Algorithm 3 describes the process of finding a $k$-element set in a symmetric deterministic diffusion model.

**Theorem 4.8.4** There exists a polynomial algorithm for the influence maximization problem for the symmetric deterministic cascade model.

**Proof** In this model, activation of a node in a graph causes activation of its neighbourhoods and this cascade process continues until none of the inactive nodes can be activated by one of the already active nodes. This fact helps us partition the graph into several partitions containing connected nodes each of which is capable of activating the whole partition. Hence, activating one of the nodes in the partition is sufficient for activation of other nodes in the partition. As stated above, this implies
that the influence sets for every node in the influence set is the influence set. Since
the k-element cover set of the graph in this model contains disjoint subsets (i.e., con-
nected components), it is sufficient to find the disjoint cover sets of the graph, greedily
pick a node from every disjoint partition according to the cardinality of the set from
which the node is picked and add the node to the set of influential nodes $A$. This
theorem is simply put in Algorithm 3.

**Example 1:** Figure 9 shows an example of applying algorithm 3 on the graph.
Threshold of all the nodes are considered equal to 0.6. The result of the algorithm is
two connected components. The result of this algorithm is $S = \{\{a, b, c, d\}, \{e, f, g\}\}$

![Diagram](image.png)

**Figure 9:** Example of applying of BFS on a graph in a non-deterministic symmetric
model; the threshold $\theta_{v_i}$ for all the nodes are considered equal to 0.6
Algorithm 2  Connected-Component algorithm which partition the network into disjoint subsets of node using BFS

Function ComponentSet BFSCC(DirectedGraph G, posting p)

VertexSet ρ // Processed set of vertices;
ComponentSet U;
i = 0;
for all v ∈ V' do
    i ← i + 1;
    if v /∈ ρ then
        VertexSet Ai = BFSδ(v, p);
        U ← U ∪ {Ai};
        ρ ← ρ ∪ Ai
    end if
end for
return U

Algorithm 3  A k-element Polynomial Influence Maximization Algorithm for Symmetric Independent Threshold Model

Function InfMaxK-Element(DirectedGraph G, posting p, k)

U ← BFSCC(G, p);
- Sort U Descendent;
for i ← 1 → k do
    vi ← RandomlyChooseANodeFrom(Ai);
    S ← S ∪ {vi}
end for
return S
4.9 Influence Maximization in Asymmetric Deterministic Cascade Model

In the asymmetric model the existence of an active edge \(< v_i, v_j >\) in a graph does not guarantee the existence of an active path in the other direction i.e., \(< v_j, v_i >\). Hence, every node generates a different influence set, these sets are neither necessarily disjoint nor equal.

Figure 11 demonstrates that by activating nodes a, b, c three sets are activated in which some nodes are in common showing the fact that nodes can be mapped to different sets which are not necessarily disjoint.

**Theorem 4.9.1** The problem of influence maximization i.e., finding a k-element set of nodes to cover the graph is NP-complete [77].

**Proof** This problem can be mapped into the cover set problem described in Chapter 2 which is an NP-complete problem. The complete proof and transformation can be found in [77].
The cardinality of the influence maximization function can be computed using the following equation:

\[
|\sigma(A)| = |\bigcup_{\forall a_j \in A} \delta(a_j)| = \\
\sum_{\forall a_j \in A} |\delta(a_j)| \\
- \sum_{\forall a_n, a_m \in A} (\delta(a_n) \cap \delta(a_m)) \\
+ \sum_{\forall a_n, a_m, a_k \in A} (\delta(a_n) \cap \delta(a_m) \cap \delta(a_k)) \\
- \ldots \\
+ \sum_{l=2}^{\infty} (-1)^{l-1} |\bigcap_{k=1}^{l} \delta(a_k)|,
\]

Figure 11: Overlap in the influence sets
4.10 Measuring Influence in non-Deterministic Cascade Model

Section 4.7 provides a low complexity method for measuring influence in a deterministic cascade model. The deterministic cascade model considers no randomness in transmission of an idea through a network. The question of interest in this thesis is measuring influence of a node considering a random process in infecting nodes across the network. In a random process, a piece of information $p_k$ may be transmitted from $v_i$ to $v_j$ according to the probability $\omega^{p_k}(v_i, v_j)$. According to Equation 36 from Section 4.4, in order to calculate the probability of activation of a node, we need to calculate all of the paths starting from a node to all the other nodes in the graph. According to [79], the problem of calculating all of the paths from one node in a graph to another node is NP-complete. This problem can be mapped into the 2-terminal network reliability problem [76].

After measuring the probabilities of activation for every node in a graph given a starting node, we need to measure the expected number of active nodes starting from the node $v_s$. This is the influence of node $v_s$. The expected number of active nodes in a graph starting from the node $v_s$ is given by the following equation:

$$E(|\delta(v_s)|) = 1 \times Pr(|\delta(v_s)| = 1)$$
$$+ 2 \times Pr(|\delta(v_s)| = 2)$$
$$+ ...$$
$$+ n \times Pr(|\delta(v_s)| = n).$$

(43)

The first part refers to the probability of activation of one node in a network which
has \(^n\choose 1\) terms as follows:

\[
Pr(\mid \delta(v_s) \mid = 1) = \sum_{\forall v_i \in V} Pr(\lambda(v_i)) \prod_{\forall v_j \neq v_i} (1 - Pr(\lambda(v_j))). \tag{44}
\]

and the probability of activation of two nodes in a network has \(^n\choose 2\) and so on. The expected number of nodes being activated starting from \(v_s\) is given by the following equation:

\[
E(\mid \delta(v_s) \mid) = 1 \times \sum_{\forall v_i \in V} Pr(\lambda(v_i)) \prod_{\forall v_j \neq v_i} (1 - Pr(\lambda(v_j)))
+ 2 \times \sum_{\forall v_i, v_j \in V} Pr(\lambda(v_i))Pr(\lambda(v_j)) \prod_{\forall v_k \neq v_i, v_k \neq v_j} (1 - Pr(\lambda(v_k)))
+ ...
+ n \times \prod_{\forall v_i \in V} Pr(\lambda(v_i)), \tag{45}
\]

which has \(^n\choose 1\) + \(^n\choose 2\) +...+ \(^n\choose n\) terms. The calculation of the above equation involves generating all of the subsets of a set which is obviously non-polynomial. The goal of this thesis is to estimate the value of the above equation or ranking nodes according to their influence in the non-deterministic cascade environment of information diffusion.

4.11 Summary

The problem of influence maximization has been studied by many researchers across the world. However, none of the previous literature concentrated on measuring influence in a non-deterministic environment. This random process can be represented as a Markov chain process. In this thesis, we concentrate on the problem of measuring influence by predicting the number of nodes activated during a chain of activations in a network.
A Markov chain model has been used in this thesis in order to measure the probability of activation of a node in a network. In the next chapter, an algorithm is proposed to measure probability of activation of all of the nodes in a network given the starting node (i.e., \( \forall v_i : Pr(\lambda^s(v_i) = 1 \mid \lambda^s(v_s) = 1) \)), and then these probabilities are used to predict the expected number of infected nodes (i.e., \( E(|\delta(v_s)|) \)).
Chapter 5

Influence Measurement Algorithms

5.1 Introduction

This chapter provides a solution to measure the capability of a node to a network in infect a larger number of nodes according to the probabilities of transmission of a new idea between all the pairs of nodes in the network. In the last chapter, it is stated that the contribution of this thesis is to measure the probability of infection of all the nodes in a network given a starting node. As this problem can be mapped into the problem of finding all of the paths between two nodes in a graph, this problem involves a non-polynomial algorithm. In this chapter, a polynomial algorithm will be proposed for measuring the probability of activation of each node given a starting point in a network. The other problem is the calculation of the expected number of infected nodes in a network given the probability of activation for all the nodes. The rest of this chapter concentrates on explaining how the proposed algorithms are utilized to measure the expected number of infected nodes. Owing to the non-polynomial time complexity of the influence problem, approximations are proposed that allow polynomial time algorithms to be utilized.
5.2 Measuring Probability of Activation in Non-deterministic Model

According to Equation 36 from Section 4.4, the probability of activation of a node in a network depends on the activation of a sequence of nodes, which is called the active path. In the next section, this problem is mapped into a Markov process in which a transition from one state to another state does not depend on the sequence of the past events.

5.2.1 Modelling Activation Process by Markov Chain Model

A Markov Chain Model describes a random process by a sequence of states characterized by a memoryless machine [34]. According to Markov Chain Theory, a transition from one state to another state depends on the current state and the probability of transition but not the sequence of events that occurred prior to reaching the current state.

Hence, we can consider the activation probability of a node in a network as a function of the probability of activation of its neighbours regardless of how those neighbours got infected. Therefore, instead of finding all of the paths between two nodes in a network, we can model the probability of the activation of a node as a function of the probability of activation of its neighbours.

In the cascade model of activation, the process is called progressive if any inactive node can turn into an active one but the process cannot happen in the other direction. However, in the non-progressive process, an active model can also become inactive.

As described above, the process of activations of a series of nodes can be represented as a sequence of transitions in a Markov chain model [34]. The non-deterministic progressive cascade model is represented as a sequence of transitions...
between $2^n$ states where $n = |\mathcal{V}|$ is the number of nodes in the network. Each state can be represented as a permutation of a binary vector of nodes $s_i = \{a_1, a_2, ..., a_n\}$ in which each element represents the activation of a node in the network. The outcome of the next experiment is independent of the series of previous states.

In each experiment, we move from one state $s_i = (a_1, ..., a_l, ..., a_k, ..., a_n)$ (in which $a_l = 1, a_k = 0$) to another state $s_j = (a_1, ..., a_l, ..., a_k, ..., a_n)$ (in which $a_l = 1, a_k = 1$) if the node $v_k$ is activated in that the value of the corresponding element $a_k$ changes from 0 to 1. This node can be activated only if there exists an already activated neighbour $v_l$ represented by $a_l = 1$ in the current state i.e., $s_i$. Hence, the transition from $s_i$ to $s_j$ happens with the probability $p_{lk} = \omega(v_l, v_k)$.

**Figure 12:** Modelling activations with Markov Chain Model

However, this is not always the case to have $2^n$ states in a network because the graph of the network is not a complete graph. In reality, the state space is much smaller owing to the limited connectivity within the network.
5.2.2 Approximating Activation probability by Iteration

In order for a node \( v_j \) to become activated, this node has to be activated by at least one of its neighbourhoods. This fact is demonstrated in the following equation:

\[
Pr(\lambda^p_k(v_j) = 1) = 1 - \prod_{\forall v_i \in \eta(v_j)} (1 - Pr(\lambda^p_k(v_i) = 1)) \times \omega^p_k(v_i, v_j),
\tag{46}
\]

The second expression in the above equation; i.e., \( \prod_{\forall v_i \in \eta(v_j)} (1 - Pr(\lambda^p_k(v_i) = 1)) \times \omega^p_k(v_i, v_j) \) is the probability that \( v_j \) is activated by none of its neighbourhoods. So, the whole expression means that \( v_j \) is activated by at least one of its neighbours. In the above equation, the probability of activation of a node is defined as a function of the probability of activation of its neighbourhoods recursively.

In order to measure the value of the aforementioned recursive equation, an algorithm is proposed similar to the power iteration method [80]. In this algorithm, the probability of activation of the node \( v_j \) at the time \( t+1 \) is a function of the probability of activation of its neighbourhoods at time \( t \). This algorithm runs in \( 0 < t < m \) steps until the difference between the value of the probabilities in two consecutive steps is less than a threshold, \( \phi \), which is an input to the algorithm. With this observation, the iterative form of Equation 46 becomes:

\[
Pr(\lambda(v_j) = 1; t + 1) = 1 - \prod_{\forall v_i \in \eta(v_j)} (1 - Pr(\lambda(v_i) = 1; t)) \times \omega(v_i, v_j),
\tag{47}
\]

Given the vector of vertex activation probabilities in the network:

\[
PR^t = \begin{bmatrix}
Pr(\lambda(v_1); t) \\
\vdots \\
Pr(\lambda(v_n); t)
\end{bmatrix}
\]
The above equation results in the following matrix manipulation where $\omega_{v_i,v_j}$ is the probability of transmission of the idea from node $v_i$ to $v_j$ and the operator $\otimes$ is defined as $PR_{j}^{t+1} = 1 - \prod_{i=1}^{N}(1 - \omega_{i,j}PR_{i}^{t})$ applied to the following matrices. $PR$ is the activation probability vector:

$$
PR_{i}^{t+1} = \begin{bmatrix}
\omega_{v_1,v_1} & \ldots & \omega_{v_1,v_N} \\
\vdots & \ddots & \vdots \\
\omega_{v_N,v_1} & \ldots & \omega_{v_N,v_N}
\end{bmatrix} \otimes PR_{i}^{t} \quad (48)
$$

### 5.2.3 Probability Propagation Algorithm

The above method can be formulated in Algorithm 4 and is called the *Probability Propagation Method*. The $\otimes$ operator is applied to the activation probability vector of the network. In every step the accumulative error is measured according to the current and previous values of the elements of the activation probability vector. This operator repeats until the accumulative error is less than a determined threshold.

**Algorithm 4** Probability propagation algorithm for measuring activation probability of nodes in a network

**Proc** Prob-Prop(double $\phi$, int $v_s.Id$, int $MI$)

$t = 0$

**while** $error > \phi$ and $t < MI$ **do**

$error=0$

$PR_{i}[v_s.Id] = 1$

$t=t+1$

$$
PR_{i}^{t+1} = \begin{bmatrix}
\omega_{v_1,v_1} & \ldots & \omega_{v_1,v_N} \\
\vdots & \ddots & \vdots \\
\omega_{v_N,v_1} & \ldots & \omega_{v_N,v_N}
\end{bmatrix} \otimes PR_{i}^{t}
$$

$error = \sum_{i}^{n} PR_{i}^{t+1}[i] - PR_{i}^{t}[i]$

**end while**

**return** $PR_{i}^{t+1}$
Since the matrix of the graph representing a social network is a sparse matrix, we apply the above method for the non-zero elements of the probability adjacency matrix. Therefore, the order of this algorithm is $|E| \times t$, where $t$ is the number of iterations and $|E|$ is the number of edges in the graph of the studied network and in the worst case is $O(|E| \times MI)$, where $MI$ is the input parameter of the system as the maximum number of iterations.

5.2.4 Convergence of The method

Conjecture 5.2.1 Algorithm 4 is convergent to the probabilities of activations when it is applied to a directed acyclic graph.

However, a graph in a social network is directed but not acyclic. In order to guarantee the convergence of the aforementioned method, the graph should be converted to an acyclic graph.

![Cycle detection](image)

**Figure 13:** Cycles in a network causes the probabilities to converge to 1 by propagation of probabilities

In order to have a directed acyclic graph, we have to remove some links from the graph by using one of the graph cycle detection algorithms or remove a link from the graph randomly, which is a naive solution. However, randomly removing a link from a graph causes missing of some paths ending to specific nodes in the graph. A method that was used in the proposed algorithm is running the algorithm iteratively
for each node while the outgoing edges have been removed from the corresponding
node. The outgoing edges makes no contribution to the probability of activation for
the corresponding node. An example of this algorithm can be observed in Figures 13
and 14.

Algorithm 5 Probability propagation algorithm for measuring activation probability
of nodes in a network; removing cycles by ignoring out going edges for a particular
node.

\begin{verbatim}
Algorithm 5 Probability propagation algorithm for measuring activation probability
of nodes in a network; removing cycles by ignoring out going edges for a particular
node.
\begin{tabular}{l}
t=0; 
Proc Prob-Prop(double $\phi$, int $v_s.Id$, int $MI$) 
for $k = 1 \rightarrow |V|$ do 
\hspace{1em} while error > $\phi$ and $t < MI$ do 
\hspace{2em} PR$^t[v_s.Id] = 1$
\hspace{2em} t=t+1 
\hspace{2em} for all $v_j \in V$ do 
\hspace{3em} for all $v_i \in \eta(v_j)$ do 
\hspace{4em} if $v_i \neq v_k$ then 
\hspace{5em} tmp = tmp * (1 - PR$^t[v_i.Id] * \omega_{v_iv_j}$) 
\hspace{4em} end if 
\hspace{3em} end for 
\hspace{2em} end for 
\hspace{2em} PR$^{t+1}[v_j.Id] = 1 - tmp$
\hspace{2em} end for 
\hspace{1em} error = $\sum_{v_i \in V} PR^{t+1}[v_i.Id] - PR^t[v_i.Id]$
\hspace{1em} end while 
return $PR_{t+1}$
\end{table}

This method increases the complexity of the algorithm by the number of nodes
in the graph; i.e., $O(|E| \times |V| \times MI)$. The new complexity is the final iterative
version of the Probability Propagation Algorithm can be seen in Algorithm 5.

5.3 Estimating Expected number of Infections

After measuring the probability of activation for every node in the network given the
starting node, we now measure the expected value of the number of infected nodes
in the network; i.e., the expected value for the cardinality of the delta function in
Figure 14: Removing cycles by eliminating outgoing edges in the graph for the nodes a,b,c,d from left to right

Equation 38. Since the process of activation of nodes in a network was modelled as a chain in an independent random process, we can assume activation of each node as an independent event with a given probability. This process is similar to flipping $n$ unfair coins with a given probability for each coin.

According to [81] we have $E(X + Y) = E(X) + E(Y)$, assuming $X, Y$ as two independent random variables. The linearity of the expected value can be generalized to $n$ independent random variables. Therefore, assuming activation of $n$ nodes in the graph as $n$ independent events, the expected value of the number of activations in the graph is a linear function of probabilities of activation; i.e., the sum of the probabilities.

Assuming activation of nodes in the graph of the corresponding network, we can imagine a direct edge from the starting node to every other node in the graph resulting in the star topology in which the probability of transition is the probability
Figure 15: Assuming a direct edge from the starting node to all the nodes in the graph

of activation of a node in the corresponding network. The following equation gives

the expected number of activations/head in $n$ independent events (i.e., node activation/unfair coin flip) with given probabilities. Equation 49 can be seen in Algorithm 6.

$$E(|\delta(v_s)|) = \sum_{\forall v_j \in V} p_{v_s,v_j}$$

(49)

Algorithm 6 Expected number of infected nodes for a particular posting in a specific social network

Proc Expected-Infected(DirectedGraph $G$, Posting $p_k$)

EI=0.0;

for all $v_i \in G.V$ do

$EI \leftarrow EI + Pr(\upsilon^p_k(v_i) = 1)$

end for

return $EI$

5.4 Summary

In this chapter a method was proposed for predicting the number of infections in a network. This method considers a series of activations as a sequence of events in a
Markov chain process. This method includes a polynomial algorithm to measure the probability of infection of a node in a network given the starting point in the network. Assuming independence among activation probabilities of the nodes in the graph of a network, it is proven that the expected number of infections in a network is the sum of probability of infection of all the nodes in the network. Algorithm 6 can be used to compute the number of infected nodes starting from each node in the network. The most influential node can therefore be determined. The next chapter will concentrate on the validation of the proposed method in this chapter using simulation of the non-deterministic independent cascade on different topologies of a network with differing number of nodes.
Chapter 6

Simulation and Experimental Evaluation

6.1 Introduction

This chapter provides experimental validation of the proposed method described in
the previous chapter for estimating the number of infected nodes in a social network.
Although many utilities and applications (e.g., Facebook, Twitter and Google+) ex-
ist for the proposed model of a social network, access to a standard real dataset is
not possible due to privacy issues in these social networks. Organizations such as
Facebook and Twitter are reluctant to release a real dataset showing diffusion of in-
formation through their networks due to the privacy concerns over their users. Hence,
we modelled a social network described in Chapter 4 and simulated the behaviours of
users in software. The following sections discuss aspects of the properties and struc-
ture of a social network as well as behaviours of individuals and contents created in
the simulated social network. In the next section, properties of a social network will
be discussed and then the structure and topologies of social networks implemented
in this project will be described. Experimental results using the proposed algorithms
from the last chapter are then provided followed by a discussion.
6.2 Simulation of a Social Network

In order to simulate a social network, we have created an infrastructure in which individuals are modelled by agents. Each agent has a number of behaviours such as creating content, and re-sharing as well as some properties such as content creation rate, re-sharing rate, vector of interests, and set of trusts to its neighbourhoods. The model which is used in this simulation is equivalent to the model described in Chapter 4.

Regarding the structure of the network, we consider a directed graph constructed using several different topologies. Evaluation of the algorithms proposed in the previous chapter have been performed using structure of real social networks (Epinion and Facebook) as well as different randomly generated topologies; i.e., scale-free network, small world and random graph network. The characteristics of this network types are described in Section 2.4.

A posting in the simulation is represented by a vector of interests \( VI_i = \langle I_{i1}, ..., I_{im} \rangle \) containing \( m \) real numbers between 0 and 1 for the agent \( v_i \). This vector is randomly generated. The post propagation rate \( PPR_i \) and post creation rate \( PCR_i \) are the properties of the agent \( i \) represented by a real number in the range of \([0, 1]\) which are randomly generated using a uniform random distribution. These properties are considered in the laziness of an agent in propagating a posting in a social network.

The wall of each agent is represented by the set \( \mathcal{W}_i = \{w_1, ..., w_l\} \) containing \( l \) elements, each is a pointer to a posting generated by the agent \( i \) itself or re-shared by the agent \( i \) from one of its friends. The elements of the wall of each agent \( i \) are accessible by its friend to consume. In this simulation, consuming a posting \( p_k \) by an agent \( v_j \) means re-sharing of the posting by adding a pointer to the wall of the agent \( v_j \) with the probability of \( \omega^{p_k}(v_i, v_j) \). More formally,
\begin{equation}
\text{Consumed}(p_k, v_i, v_j) = \begin{cases} 
W_j = W_j \cup \{p_k\} & \text{if } \Gamma_{pk}(v_i, v_j) = 1 \\
\text{do nothing} & \text{Otherwise}
\end{cases}
\end{equation}

The trust set $T^k_i = \{T^k_{i1}, \ldots, T^k_{in}\}$ is also constructed by generating its elements using a Gaussian random generator each of which represents the degree of trust of the agent $v_i$ has in its friend, agent $v_j$ on the topics related to the posting $p_k$. It should be stressed here that the simulation represents a snapshot of the time evolution of the social network and, as such, trust does not change during the simulation. It is left as future work to include trust models as described in Section 3.4 of the thesis.

### 6.3 Behaviours in a Social Network

Behaviour in a social network is defined as how users interact with each other. These interactions include: creating content (i.e., posting) by an individual and sharing it with its friends, re-sharing of the contents shared by an individuals’ friends, submitting comments or liking friends’ postings. These behaviours have been formally defined in Chapter 4. Here, we describe how these behaviours are simulated in order to validate algorithms proposed in the previous chapter. As this thesis focuses on the models of information diffusion and algorithms for anticipating number of infected nodes in a network and the capability of individuals in spreading information through the network, the first two aforementioned behaviours i.e., posting creation and re-sharing are simulated.

#### 6.3.1 Simulation of Posting Creation

In this simulation, a posting $p_k$ is represented by the vector of relatedness $VT_k = \langle t_{k1}, \ldots, t_{km} \rangle$. Each element of this vector is generated according to the corresponding
element in the vector of the interest of the user \( v_i \). Hence, a number \(-1 < d_i < 1\) is randomly generated as noise according to the normal distribution and is added to the corresponding element of the vector of relatedness. In other words, \( \forall t_{kl} \in VT_k : \)

\[
t_{kl} = \begin{cases} 
    I_{il} + d_i & 0 \leq I_{il} + d_i \leq 1 \\
    0 & I_{il} + d_i < 0 \\
    1 & I_{il} + d_i > 1 
\end{cases}
\]  

(51)

where \( I_{il} \) is the \( l^{th} \) element in the vector of interest of the user \( v_i \).

According to the blue line in Figure 16 following a normal distribution with \( \mu = 0, \sigma = 0.2 \), a random number is more likely to be close to 0 because the probability of generating a number close to 0 is higher than the probability of generating other numbers in the range of \([-1, 1]\). Therefore, the noise generated by the normal distribution causes the the created post to be close to the topics of interests of agent \( v_i \). The smaller the value of \( \sigma \) is, the closer topics of the posting to the interest of the user will be.
6.3.2 Simulation of Re-Sharing of Postings

In Chapter 4, two categories of cascade models have been defined. These models have been implemented in the simulation of the network and the results are shown in the next sections. As the contribution of this thesis is a solution for measuring the probability of propagation of a posting in a social network in a non-deterministic model, this model is described in the following section. However, for the deterministic model an algorithm was proposed and proved to work in Chapter 4. Therefore, here, we concentrate on the non-deterministic model.

For the simulation of the non-deterministic cascade model formulated in Section 4.3.2 in Chapter 4, after adding the posting to the wall of the starter agent, the agent gets infected and activated. Afterwards, each of its neighbours gets a single chance to propagate the posting according to the probability \( \omega_{s}(v_s, v_i) \). This process can be simulated by generating a random real number \( 0 \leq d \leq 1 \) with a uniform distribution and comparing the value of \( d \) to the probability of the propagation. If the probability of propagation exceeds the value of random variable \( d \), the post will be added to the wall of the target agent \( v_i \) and the agent will be infected and activated as well to propagate the posting.

\[
\Lambda^{p_k}(v_s, v_i) = \begin{cases} 
1 & d \leq \omega^{p_k}(v_s, v_i) \\
0 & Otherwise 
\end{cases}
\]  

(52)

6.3.3 Probability of Re-sharing of a Posting

As stated in Section 4.3.3 in Chapter 4, the probability of propagation of a posting from the user \( v_i \) to \( v_j \) is a function of the similarity between the vector of interests of agent \( v_j \) and the vector of related topics of posting \( p_k \), the laziness of the target agent \( v_j \) and the trust value between two agents. More formally,
\[ \omega^{pk}(v_i, v_j) = f(\Psi(v_j, p_k), \tau^{pk}(v_j, v_i), \Gamma(v_j)). \]

For the experimental evaluation performed in this thesis, the function \( f \) has been implemented as a simple product of these three functions. The similarity is calculated using Pearson’s correlation coefficient, trust is a random number assigned to each relation, and the laziness is also a random number generated by a uniform distribution and assigned to each agent as a property of the agent. Therefore, we have:

\[ \omega^{pk}(v_i, v_j) = \Psi(v_j, p_k) \times \tau^{pk}(v_j, v_i) \times \Gamma(v_j). \] (53)

where \( \Psi(v_j, p_k) = \text{Correl}(v_j.VI, p_k.VT) \).

### 6.4 Simulation Algorithms

Algorithm 7 demonstrates the simulation of content re-sharing in a non-deterministic cascade environment.

**Algorithm 7 Simulation of consuming a content in a Non-Deterministic Cascade Model**

```plaintext
Proc Consume(Posting p, Node v_i, Node v_j)
  \[ d = \text{rand}(); \]
  \[ \text{probability} = \text{Correl}(v_j.VI, p.VT) \times \tau^p(v_j, v_i) \times \Gamma(v_j); \]
  \[ \text{if } d < \text{probability} \text{ then} \]
    \[ v_j.W = v_j.W \cup \{p\} \]
    \[ \text{for all } v_k \in \eta(v_j) \text{ do} \]
      \[ \text{Consume}(p, v_j, v_k); \]
    \[ \text{end for} \]
  \[ \text{end if} \]
```

In the implementation of the simulation software, the main class is called agent. This class extends thread class and contains all of the behaviours of an agent implemented in different functions. Therefore, for each individual in the simulated social network, a thread as an instance of the class is created. Each instance of the agent class gets its properties from database which stores structure of a network as well as
behavioural parameters of an agent (PPR, PCR, Vector of interests, Laziness, and Trust values between agents). In this simulation each thread represents an individual in social network. Each thread generate a posting according to its interests and its behavioural properties and the posting is added to the queue of his neighbourhoods to be consumed regarding Algorithm 7.

A component called Jung Framework [83] has been used to generate randomly generated graph structures i.e., small-world network, random graph, and scale-free network. This component is usually used for graph visualization.

6.5 Experimental Evaluation

The suggested algorithm in the previous section has been validated using a simulation. In the experimental results, several topologies with varying number of nodes have been tested to validate the suggested method. The algorithm was tested on Epinion, Facebook, random graph, small-world, scale free network with three scales of number of nodes: 10, 100 and 1000 nodes. Epinion and Facebook topologies were created from the dataset downloaded from the Stanford Network Analysis Project website [84,85]. The following table provides details on these datasets. In this research, for each topology, one instance of the network was created using sampling of 10, 100 and 1000 nodes and selecting the corresponding edges for the sampled nodes. In the simulation, 1 posting was generated by one of the nodes randomly, and that posting was tested on all the nodes to measure the number of infected nodes for 1000 iterations. Therefore, the number of times that a specific node is infected starting from a given node divided by 1000 represents the probability of infecting that node starting from the given node. Formally,

$$P_{P_{p_{s}}}^{p_{s}}(\lambda(v_{i}) \mid \lambda(v_{s})) = \frac{\# \text{ of times the node } v_{i} \text{ gets infected}}{\# \text{ of Iteration} = 1000}. \quad (54)$$
<table>
<thead>
<tr>
<th>Dataset</th>
<th># of Nodes</th>
<th># of Edges</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epinion</td>
<td>75,879</td>
<td>508,837</td>
<td>Directed</td>
<td>Who-trusts-whom of Epinions.com</td>
</tr>
<tr>
<td>Facebook</td>
<td>63,891</td>
<td>274,076</td>
<td>Directed</td>
<td>Friendship</td>
</tr>
</tbody>
</table>

**Table 3:** Structural properties of Epinion and Facebook datasets

As stated in the above, the Random-graph, small-world and scale-free networks were generated using a component called *Jung Framework* [83]. The properties of these networks are shown in Table 4 including number of nodes, number of edges, maximum degree in the graph, average degree in the graph. The pro/coef column of Table 4 contains uniformation specific to a particular network type. For random graphs this column determines the probability of existence of an edge between two nodes. The maximum and average number of propagations in different networks are demonstrated in Tables 5 and 6. Please note that the threshold $\phi$ for accumulated error in *Probability Propagation Algorithm* (i.e., Algorithm 5) was set to 0.001 in all of the experiments.

The results of these simulations are shown in Table 7. Spearman and Pearson correlation coefficients were used to measure the correlation between the anticipated number of nodes infected in the network and the results of simulation on different networks. In fact, the expected number of nodes calculated by the proposed algorithm is a rank which sorts nodes according to their capability of infecting higher numbers of nodes in a network.

The graphs in Figure 18 show the relationship between the result of the proposed probability propagation algorithm and the simulation of the corresponding network for 10, 100 and 1000 nodes simulated on a specific node in the Epinion network. Figure 19 shows the relationship between number of nodes and number of edges and maximum degree in different networks. The graphs in Figure 20 shows the
<table>
<thead>
<tr>
<th>Dataset</th>
<th># of Nodes</th>
<th># of Edges</th>
<th>Max degree</th>
<th>Average degree</th>
<th>Prob/Coef</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epinion</td>
<td>10</td>
<td>13</td>
<td>6</td>
<td>2.6</td>
<td>N/A</td>
</tr>
<tr>
<td>Facebook</td>
<td>10</td>
<td>6</td>
<td>3</td>
<td>1.2</td>
<td>N/A</td>
</tr>
<tr>
<td>Random Graph</td>
<td>10</td>
<td>5</td>
<td>3</td>
<td>1</td>
<td>0.1</td>
</tr>
<tr>
<td>Small World Network</td>
<td>10</td>
<td>26</td>
<td>7</td>
<td>5.2</td>
<td>0.5</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>10</td>
<td>10</td>
<td>3</td>
<td>1.9</td>
<td>N/A</td>
</tr>
<tr>
<td>Epinion</td>
<td>100</td>
<td>1359</td>
<td>95</td>
<td>27.18</td>
<td>N/A</td>
</tr>
<tr>
<td>Facebook</td>
<td>100</td>
<td>63</td>
<td>5</td>
<td>1.2</td>
<td>N/A</td>
</tr>
<tr>
<td>Random Graph</td>
<td>100</td>
<td>239</td>
<td>10</td>
<td>4.78</td>
<td>0.05</td>
</tr>
<tr>
<td>Small World Network</td>
<td>100</td>
<td>300</td>
<td>8</td>
<td>6</td>
<td>0.5</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>100</td>
<td>100</td>
<td>15</td>
<td>1.99</td>
<td>N/A</td>
</tr>
<tr>
<td>Epinion</td>
<td>1000</td>
<td>29509</td>
<td>486</td>
<td>59.01</td>
<td>N/A</td>
</tr>
<tr>
<td>Facebook</td>
<td>1000</td>
<td>2184</td>
<td>44</td>
<td>4.36</td>
<td>N/A</td>
</tr>
<tr>
<td>Random Graph</td>
<td>1000</td>
<td>10022</td>
<td>36</td>
<td>20.04</td>
<td>0.02</td>
</tr>
<tr>
<td>Small World Network</td>
<td>1000</td>
<td>2882</td>
<td>10</td>
<td>5.76</td>
<td>0.5</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>1000</td>
<td>1000</td>
<td>42</td>
<td>1.99</td>
<td>N/A</td>
</tr>
<tr>
<td>Epinion</td>
<td>10000</td>
<td>216191</td>
<td>1944</td>
<td>43.23</td>
<td>N/A</td>
</tr>
<tr>
<td>Facebook</td>
<td>10000</td>
<td>36150</td>
<td>116</td>
<td>7.23</td>
<td>N/A</td>
</tr>
<tr>
<td>Random Graph</td>
<td>10000</td>
<td>998346</td>
<td>250</td>
<td>199.66</td>
<td>0.001</td>
</tr>
<tr>
<td>Small World Network</td>
<td>10000</td>
<td>29999</td>
<td>11</td>
<td>5.99</td>
<td>0.5</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>10000</td>
<td>10000</td>
<td>98</td>
<td>2</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**Table 4:** Properties of the networks used in the simulation. The last column shows the probability of existence of an edge in a Random-Graph or clustering coefficient in a small-world network according to which random graphs or small-world networks were generated.
<table>
<thead>
<tr>
<th>Dataset</th>
<th># of Nodes</th>
<th># of Edges</th>
<th>Max # of Infected Nodes</th>
<th>Max # of Hops</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epinion</td>
<td>10</td>
<td>13</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>Facebook</td>
<td>10</td>
<td>6</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>Random Graph</td>
<td>10</td>
<td>5</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Small World Network</td>
<td>10</td>
<td>26</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>10</td>
<td>10</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Epinion</td>
<td>100</td>
<td>1359</td>
<td>29</td>
<td>15</td>
</tr>
<tr>
<td>Facebook</td>
<td>100</td>
<td>63</td>
<td>11</td>
<td>6</td>
</tr>
<tr>
<td>Random Graph</td>
<td>100</td>
<td>239</td>
<td>25</td>
<td>14</td>
</tr>
<tr>
<td>Small World Network</td>
<td>100</td>
<td>300</td>
<td>20</td>
<td>10</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>100</td>
<td>100</td>
<td>14</td>
<td>4</td>
</tr>
<tr>
<td>Epinion</td>
<td>1000</td>
<td>29509</td>
<td>275</td>
<td>139</td>
</tr>
<tr>
<td>Facebook</td>
<td>1000</td>
<td>2184</td>
<td>83</td>
<td>37</td>
</tr>
<tr>
<td>Random Graph</td>
<td>1000</td>
<td>10022</td>
<td>299</td>
<td>96</td>
</tr>
<tr>
<td>Small World Network</td>
<td>1000</td>
<td>2882</td>
<td>66</td>
<td>27</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>1000</td>
<td>1000</td>
<td>42</td>
<td>7</td>
</tr>
</tbody>
</table>

**Table 5:** Statistics about the maximum number and level of propagations and hops to which a posting penetrated
<table>
<thead>
<tr>
<th>Dataset</th>
<th>% of Nodes</th>
<th>% of Edges</th>
<th>Avg % of infected nodes</th>
<th>Avg % of Hops</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epinion</td>
<td>10</td>
<td>13</td>
<td>2.0</td>
<td>0.69</td>
</tr>
<tr>
<td>Facebook</td>
<td>10</td>
<td>6</td>
<td>1.75</td>
<td>0.5</td>
</tr>
<tr>
<td>Random Graph</td>
<td>10</td>
<td>5</td>
<td>2.0</td>
<td>0.8</td>
</tr>
<tr>
<td>Small World Network</td>
<td>10</td>
<td>26</td>
<td>2.19</td>
<td>0.99</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>10</td>
<td>10</td>
<td>1.72</td>
<td>0.51</td>
</tr>
<tr>
<td>Epinion</td>
<td>100</td>
<td>1359</td>
<td>4.2</td>
<td>1.75</td>
</tr>
<tr>
<td>Facebook</td>
<td>100</td>
<td>63</td>
<td>1.97</td>
<td>0.73</td>
</tr>
<tr>
<td>Random Graph</td>
<td>100</td>
<td>239</td>
<td>3.23</td>
<td>1.43</td>
</tr>
<tr>
<td>Small World Network</td>
<td>100</td>
<td>300</td>
<td>2.19</td>
<td>0.93</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>100</td>
<td>100</td>
<td>2.24</td>
<td>0.72</td>
</tr>
<tr>
<td>Epinion</td>
<td>1000</td>
<td>29509</td>
<td>21.39</td>
<td>5.24</td>
</tr>
<tr>
<td>Facebook</td>
<td>1000</td>
<td>2184</td>
<td>9.64</td>
<td>2.96</td>
</tr>
<tr>
<td>Random Graph</td>
<td>1000</td>
<td>10022</td>
<td>4.81</td>
<td>1.96</td>
</tr>
<tr>
<td>Small World Network</td>
<td>1000</td>
<td>2882</td>
<td>3.1</td>
<td>1.3</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>1000</td>
<td>1000</td>
<td>1.85</td>
<td>0.51</td>
</tr>
</tbody>
</table>

**Table 6:** Statistics about the average number and level of propagations and hops to which a posting penetrated
<table>
<thead>
<tr>
<th>Dataset</th>
<th># of Nodes</th>
<th>Pearson Correlation</th>
<th>Spearman’s Rank Correlation</th>
<th>Pearson correlation for Probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epinion</td>
<td>10</td>
<td>0.98</td>
<td>0.98</td>
<td>0.98</td>
</tr>
<tr>
<td>Facebook</td>
<td>10</td>
<td>0.99</td>
<td>0.98</td>
<td>0.99</td>
</tr>
<tr>
<td>Random Graph</td>
<td>10</td>
<td>1.0</td>
<td>1.0</td>
<td>1.00</td>
</tr>
<tr>
<td>Small World Network</td>
<td>10</td>
<td>0.99</td>
<td>0.95</td>
<td>0.99</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>10</td>
<td>0.99</td>
<td>0.98</td>
<td>0.99</td>
</tr>
<tr>
<td>Epinion</td>
<td>100</td>
<td>0.96</td>
<td>0.97</td>
<td>0.94</td>
</tr>
<tr>
<td>Facebook</td>
<td>100</td>
<td>0.89</td>
<td>0.98</td>
<td>0.70</td>
</tr>
<tr>
<td>Random Graph</td>
<td>100</td>
<td>0.94</td>
<td>0.97</td>
<td>0.89</td>
</tr>
<tr>
<td>Small World Network</td>
<td>100</td>
<td>0.98</td>
<td>0.98</td>
<td>0.98</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>100</td>
<td>0.90</td>
<td>0.96</td>
<td>0.81</td>
</tr>
<tr>
<td>Epinion</td>
<td>1000</td>
<td>0.85</td>
<td>0.86</td>
<td>0.80</td>
</tr>
<tr>
<td>Facebook</td>
<td>1000</td>
<td>0.73</td>
<td>0.91</td>
<td>0.47</td>
</tr>
<tr>
<td>Random Graph</td>
<td>1000</td>
<td>0.96</td>
<td>0.97</td>
<td>0.86</td>
</tr>
<tr>
<td>Small World Network</td>
<td>1000</td>
<td>0.37</td>
<td>0.86</td>
<td>0.34</td>
</tr>
<tr>
<td>Scale Free Network</td>
<td>1000</td>
<td>0.73</td>
<td>0.91</td>
<td>0.75</td>
</tr>
</tbody>
</table>

**Table 7:** Correlation between anticipated number of infected nodes by the proposed algorithm and results of simulation; the last column shows correlation between probability of activation of nodes in a network and probability of activation for the nodes in the simulation.
relation between number of nodes and average number of infected nodes and average number of hops that a posting penetrated. Finally, Figure 21 shows the trend in the relationship between correlation of the proposed algorithm and simulation of propagation of posting in different networks.

6.6 Discussion

The first observation on the results clearly shows that the behaviour of the proposed algorithm is different on different topologies. According to Table 4 it is observed that the average degree in scale-free and small world network is constant as the number of nodes increases. In the case of a small world network, the maximum degree is constant with a slightly increasing trend. However, in scale-free networks, it is observed that the maximum degree dramatically rises when the number of nodes in the graph increases. It increases the gap between maximum degree and average degree in a scale-free network.

On the other hand, in a non-deterministic process, for example in flip coin experiment, the number of heads is directly related to the number of trials. Since, by increasing average degree in a graph, the number of trials increases; therefore, the average degree has a direct relationship with the number of infected nodes in our simulation. Except scale-free network, in all of the networks, the number of nodes has an increasing trend and as a result the number of propagations (i.e., number of infected nodes) has a similar trend in this simulation. This fact can be derived from Table 6. The average number of hops to which the posting penetrated is demonstrated to be constant in most of the networks on which the simulation ran, except Epinion and Facebook. In Epinion and Facebook, the average number of hops and average number of infected nodes increases when the number of nodes increases in the graph while this number is almost constant for the generated graphs; i.e., scale-free,
Figure 17: Correlation between probability in simulation and probability propagation algorithm for 10 and 100 nodes
Figure 18: Correlation between probability in simulation and probability propagation algorithm for 1000 nodes
Figure 19: Relation between number of nodes and number of edges in different networks and maximum degree in these networks
Figure 20: Average number of infected nodes, average number of hops
Figure 21: Relation between Spearman’s correlation and number of nodes

small world network and random graph. It can be observed that the average number of hops linearly increases in random graphs. It is also observed that the maximum degree is almost equal in Epinion and random graph while maximum number of hops and maximum number of infected nodes has a similar value in these two networks.

The empirical evaluation of the Probability Propagation algorithm shows a high value of correlation between the results of this algorithm and the simulation of propagation in a social network. This correlation was measured by Pearson Correlation Coefficient. It is observed that this correlation decreases when the size of a network grows. Specifically, this correlation decreases dramatically in Facebook and small world network. It is also observed that the exact value of the probability of activation estimated by the Probability Propagation algorithm is very close to the results of simulation in a network with a small number of vertices. However, by increasing
the number of nodes, the difference between the results of Probability Propagation algorithm and simulation increases causing a small gap between two graphs with 1000 nodes as demonstrated in Figure 18.

The final evaluation of the proposed algorithm is measuring the expected number of infected nodes. This metric shows the influence of a node in a network. The proposed method was evaluated by Pearson Correlation and Spearman’s Rank coefficient. While we are interested in the order of influence, Spearman’s rank coefficient can appropriately measure the relationship between this order and the number of infected nodes in the simulation. Generally, the correlation between the proposed algorithm and the results of the simulation has a decreasing trend with increasing size of the graph. This falling trend in correlation between the proposed algorithm is more strongly observed in a small world network and Facebook.

6.7 Summary

In summary, this chapter described a method for validating the results of the algorithms from Chapter 5. Specific details of the simulation of a social network were described. The simulation includes behavioural analysis using a non-deterministic independent cascade model and different network generating algorithms. In the next step, the suggested method for measuring influence of a node in a network, was implemented and tested on the generated dataset. Finally, the correlation between the proposed method and simulation of a network was measured. The results of simulation empirically demonstrate that the suggested method is effective predicting the number of infected nodes and the probability of infection of all of the nodes using the starting point in a given network.
Chapter 7

Conclusion and Future Work

7.1 Introduction

This thesis discussed the problems that exist in traditional advertising systems and traditional on-line recommender systems. According to previous research, on the one hand these systems suffer from lack of attention by on-line customers because of their tendency to bombard users with a large number of non-useful advertisements and spam. On the other hand, traditional advertising systems are not capable of recognizing users’ needs in a timely manner. In order to solve these problems, we need to define a new strategy for recommending products to customers.

Social recommender systems are a new generation of on-line advertising systems which provide solutions for the problems of lack of attention from the target customers and recognizing users’ needs at the appropriate time. An important infrastructure for these systems is a social network in which individuals and their relations are explicitly represented. One of the problems in social advertising systems is measuring influence of individuals in the network. This problem was formally defined and an algorithm was proposed for its measurement in this thesis. This chapter describes the contributions of this thesis as well as future work related to maximizing influence in social networks.
7.2 Summary of Contributions

This thesis provides information about disadvantages of traditional recommendation systems and then describes the motivation behind a new generation of these systems called social recommender systems. It is discussed that there are many problems that need to be solved in implementing these systems. One of these problems is determining the right person in a social network according to his/her capability in the propagation of information about a new product or a service. This problem was formally described in this thesis.

In Chapter 3, related work and a literature review is provided. Chapter 3 discusses other research in the area of social network analysis and related concepts such as similarity, trust and intention of people in using social media. According to the Bass Model, we discussed that a large number of technology adaptations are performed in a large portion of society who adapt to a new innovation. Chapter 3 also provides considerable information regarding structural and behavioural social network analysis in addition to an introduction to different diffusion models in social networks. Finally, Chapter 3 provides information about time evolution of structures in social networks.

Chapter 4, after defining a formal model for an specific social network, discusses how a combination of structural and behavioural analysis is a more effective way for analysis of behaviours of users in propagation of information in a network. Chapter 4 formally describes the problem of measuring influence as well as the problem of maximizing influence in a social network according to propagation of information in a network. This chapter also formulated the behaviour of individuals in a social network in a formal model. Following this, we provided two diffusion models for describing a process of propagation of information in a network of individuals; i.e., the deterministic model and non-deterministic model. A polynomial time algorithm for measuring and maximization of influence in the symmetric deterministic model
were proposed and the soundness and completeness of these algorithms were proved. In a symmetric deterministic model, both measuring and maximization of influence can be solved in polynomial time. However, in the asymmetric model, the problem of influence maximization is NP-complete. It is also proven that the problem of measuring influence in the non-deterministic model is NP-complete in addition to the included theorems for definition of this problem in symmetric model vs. asymmetric model.

In addition to the deterministic model defined in this thesis, we provided algorithms for measuring influence in the non-deterministic model. Chapter 5 provided an algorithm to measure probability of infection of nodes in the network according to a method called Probability Propagation. This method uses the notion of a Markov chain process to describe the process of activation of nodes in a network. This algorithm consists of a combination of two concepts: Markov chain theory and eigen vector centrality. Afterwards, the results of this algorithm is used to estimate the expected number of nodes assuming independence among activation of nodes in the given network.

Chapter 6 describes the process of validation of the probability propagation method by running a simulation over different structures of a network. This chapter provides details of the simulation for validation of the proposed method and discussed the correlation between the results of simulation of a social network and the estimated influence rank by the proposed algorithm. It is observed that the proposed method can estimate the number of activated nodes in a network given the starting point for propagation with high level of accuracy. Chapter 6 also provides an infrastructure for simulation of a social network.
7.3 Future Work

This thesis provided a method for measuring influence of users in a social network at a snapshot in time. One of the most important aspects of this research is mining real data from live datasets. As stated in the previous chapters, the proposed algorithms were validated using a simulation environment for a social network. However, the next step is applying these algorithms on a real dataset. Due to the privacy issue, this work was not possible in this thesis. However, the next step is mining real data and measuring the probability of transmission between different nodes in a network according to real data set such as Facebook might provide.

One potential future work in this research is extraction of the probability of transmissions of a posting from one node to its neighbourhoods in a live dataset which involves content analysis and text mining tasks. In this approach, we need to analyze the contents shared in a social network, categorize content, cluster contents and finally discover the hidden knowledge in the raw data.

One extension required to the work undertaken in this thesis is using trust models in order to update trust values of an individual according to his/her previous interactions. The network then has a dynamic nature as friends are dropped and new ones are added. Hence, the value of trust between users is changed during their interaction in the real world. A model is required to update these values according to the history of interrelationship actions; defection or cooperation. As an example, sharing a spam link or spam information in a network can be considered as defection and sharing a useful information can be considered as cooperation. These models are described in Chapter 3.

Continuing with the view that the network is a time evolving entity, the evolution process applies not only on the behaviour of individuals in a network but also affects the structure of the network. For example, laziness should be dependent upon time.
Individuals in networks remain active for periods of time then exhibiting a refractory period. The structure of a network is also evolves by cutting or adding a relation between two individuals or a new individual joining the network.

Other possible topics for research is using the proposed algorithm for measuring influence of a user and utilizing measured probabilities of activation for each node in the influence maximization problem (i.e., covering vertices problem). There exist several heuristic algorithms for cover set problems. The proposed approach in this thesis can be utilized as a heuristic in one of the heuristic search algorithm to find the best covering k-subset.
List of References


