On the Interaction of Influence and Trust in Social Networks

Behnam Hajian and Tony White

School of Computer Science, Carleton University
1125 Colonel By Drive, Ottawa, Ontario, K1S 5B6 Canada
{bhajian, arpwhite}@scs.carleton.ca

Abstract. This paper focuses on finding important nodes in a social network based on their behaviour as well as the structure of the network; a problem of considerable interest in social recommender systems. This paper views the propagation of information in a social network as a process of infection. Furthermore, a proposal regarding infection and trust is made. The paper 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. Assuming independence in activation of network nodes, 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 information diffusion model is simulated on several classes of network.

1 Introduction

On-line recommender systems are disregarded or ignored by their users generally because their users feel suggestions from these systems are useless [6]. It has been empirically demonstrated that the main reason for this feeling is that people prefer to get recommendations from the people they trust rather than automated advertising software [3].

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.

It has been observed that people are infected by a new idea from being exposed to the innovation by their friends, rather than advertisements. This fact motivates marketers to turn to viral marketing rather than ad-hoc advertising. It is clear that influence varies from node to node in a network and identifying the most influential nodes can significantly affect the effectiveness of an advertising campaign. It is this problem that motivates the research reported in this paper.

This paper focuses on the problem of ranking nodes according to their ability to infect more nodes if that node individually gets infected. This problem is referred to in this paper as the influence measurement problem because influence is modelled as the ability of a node to infect other nodes in the network. The
process of infection is considered to affect trust between two agents: infection increases trust, failure to infect decreases it.

The remainder of this paper is organized in five sections. Section two describes related work. In Section three, a model of information diffusion is proposed in the context of a social network. Section four provides proposed algorithms for measuring influence in a social network. In order to validate the proposed algorithms, a general framework of a social network is simulated. Section five discusses the properties of our simulations, experimental results and includes a discussion of the results. Finally, conclusions and future work will be discussed in Section six.

2 Related Work

There are several methods for measuring node importance in networks; such as: Closeness, Betweenness, and Eigenvector centrality. Simply put, these metrics concentrate on the structure of the network rather than the behavior of nodes and their interactions. However, researchers have also considered other aspects of analysis, such as analysis of users’ behavior in a network. Some researchers consider a weight for each tie that shows the importance and strength of a link between two users based on their historical interactions. These are described in the following paragraphs.

Meeyoung Cha et al. [2] studied the measurement of influence in Twitter based on the following three metrics:

- 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.

Meeyoung Cha et al. [2] used Spearman’s rank correlation coefficient for comparing users’ influence. 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. Meeyoung Cha et al. also revealed that influence is not gained accidentally but requires that users need to be consistently active in the network, an observation corroborated by Khrabrov and Cybenko [10].

Afrasiabi and Benyoucef [1] studied influence as a function of link strength and incoming and outgoing clustering value defined for each node in the network. 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. They filter the spam and inactive nodes according to their activities and their interaction with other users.

Kempe et. al. [9] modelled influence by two basic diffusion models; i.e., Linear Threshold and Independent Cascade in which nodes are categorized as active or inactive nodes. The former model lies at the core of most subsequent generalizations. In the Linear Threshold model, a node \( v \) is influenced by each neighbourhood \( w \) with a weight \( b_{v,w} \) such that \( \sum_{\text{w in neighborhood of } v} b_{v,w} \leq 1 \). In order for \( v \) to become active, the sum of the weights of its active neighbours should be greater than a given threshold \( \theta_v \). The Independent Cascade model is based on interacting particle systems from probability theory in which a node \( u \) may become active at time \( t + 1 \) according to the probability \( p_{u,v} \) if \( v \) had become active in the time \( t \). This process runs until no more activations are possible. Kempe et al. proved that both of the aforementioned models are sub-modular. Finally, it is proven that the k set influence maximization problem is NP-hard for the aforementioned models. They have also used approximation methods to estimate influence maximization.

Dynamic Graph Analysis has been studied by Khrabrov and Cybenko [10] in which the number of daily mentions for each user is considered as a indicator for computing different ranks such as PageRank, drank, and starrank for influence analysis of each node in a network. For example, starrank considers user importance with respect to his or her neighborhoods. These researchers used several primitive indices in combination, such as Contiguous Longest Increasing Subsequences (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.

### 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. 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 remains 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.

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

(1)

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.

3.1 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,

\[
A_{p^k}(v_i,v_j) = \begin{cases} 
1 & \zeta_{p^k}(v_i,v_j) = 1 \text{ and } d < \omega_{p^k}(v_i,v_j) \\
0 & \text{Otherwise}
\end{cases}
\] (2)

3.2 Probability of Transmission

The probability that a piece of information \(p_k\) is transferred from the node \(v_i\) to the node \(v_j\) depends on several factors. In this paper, the following factors are considered as determining parameters of this process:

- **Similarity** 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 \(p_k\).

\[
\Psi(v_j,p_k) = sim(v_j.VI, p_k.VR),
\] (3)

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

- The **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. This parameter is a time varying parameter updated according to a trust model based upon whether a posting is propagated or not.

- **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 \(PPR_j\) [14].

Now, we define the probability of transmission of the posting \(p_k\) from user \(v_i\) to \(v_j\). The function \(\omega_{p^k}(v_i,v_j)\) returns the aforementioned probability. The omega function in the equation below is the capability of \(v_i\) in infecting \(v_j\):
\[ \omega^p_k(v_i, v_j) = \Pr(A^p_k(v_i, v_j) \mid \zeta^p_k(v_i, v_j)) = f(\Psi(v_j, p_k), \tau^p_k(v_j, v_i), \Gamma(v_j)), \] (4)

3.3 Activation Probability of a Node in a Network

According to [9], 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 one another’s attempts.

As shown in Figure 1, 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 the probability of transmission from \( v_i \) to \( v_j \). In other words:

\[ \Pr(A^p_k(v_i, v_j) = 1) = \Pr(\zeta^p_k(v_i, v_j) = 1) \times \omega^p_k(v_i, v_j), \] (5)

**Definition** The path \( path^p_k(v_s, v_j) \) is called an active path iff \( \forall v_n, v_m \in path^p_k(v_s, v_j) : A^p_k(v_n, v_m) = 1 \) then the edge \( < v_n, v_m > \) is called an active edge.

**Fig. 1:** Multi-path transmission in a network

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

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.

### 3.4 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 \(v_s\).

\[
\delta(v_s) = \eta(v_s) \cup \bigcup_{v_l \in \eta(v_s)} \delta(v_l),
\]

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. Formally, a node \(v_i\) is the most influential node in the network iff:

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

The above equation models influence as a recursive concept that considers influence of a node as a function of influence of its neighbourhoods which is similar to [5].

### 4 Algorithm for Measuring Influence in a Social Network

The non-deterministic progressive cascade model is represented as a sequence of transitions Markov chain model [13] between \(2^n\) states where \(n = |V|\) 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 \(p_{lk} = \omega(v_l, v_k)\).
4.1 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:

$$Pr(\lambda_p(v_j) = 1) = 1 - \prod_{v_i \in \eta(v_j)} (1 - Pr(\lambda_p(v_i) = 1) \times \omega_p(v_i, v_j)), \quad (9)$$

The second expression in the above equation; i.e., $\prod_{v_i \in \eta(v_j)} (1 - Pr(\lambda_p(v_i) = 1) \times \omega_p(v_i, v_j))$ 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 order to measure the value of the recursive equation above, an algorithm is proposed similar to the power iteration method [7]. In this algorithm, the probability of activation of the node $v_j$ at 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 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 9 becomes:

$$Pr(\lambda(v_j) = 1; t + 1) = 1 - \prod_{v_i \in \eta(v_j)} (1 - Pr(\lambda(v_i) = 1; t) \times \omega(v_i, v_j)), \quad (10)$$

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 $PR^{t+1} = 1 - \prod_{i=1}^N (1 - \omega_{i,j} PR^t_i)$ applied to the following matrices. $PR$ is the activation probability vector:

$$PR = \begin{bmatrix} \omega_{v_1,v_1} & \cdots & \omega_{v_1,v_N} \\ \vdots & \ddots & \vdots \\ \omega_{v_N,v_1} & \cdots & \omega_{v_N,v_N} \end{bmatrix} \otimes PR^t \quad (11)$$

4.2 Probability Propagation Algorithm

The above method can be formulated in Algorithm 1 called the Probability Propagation Method. 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 $O(|E| \times |V| \times MI)$ where $MI$ is the maximum number of iterations, a parameter input to the system. Assuming activation of nodes in the graph according to Algorithm 1, we can imagine a direct edge from the starting node to every other node in the graph. The following equation then gives the expected number of activations:

$$E(|\delta(v_s)|) = \sum_{v_j \in V} p_{v_s,v_j} \quad (12)$$
Algorithm 1 Probability propagation algorithm for measuring activation probability of nodes in a network; removing cycles by ignoring out going edges for a particular node

\[ t = 0 \]

\[ \text{Proc Prob-Prop}(\text{double } \phi, \text{ int } v_s.Id, \text{ int } MI) \]

\[ \text{for } k = 1 \rightarrow |V| \text{ do} \]

\[ \text{while } \text{error} > \phi \text{ and } t < MI \text{ do} \]

\[ PR^t[v_s.Id] = 1 \]

\[ t = t + 1 \]

\[ \text{for all } v_j \in V \text{ do} \]

\[ tmp = 1 \]

\[ \text{for all } v_i \in \eta(v_j) \text{ do} \]

\[ \text{if } v_i \neq v_k \text{ then} \]

\[ tmp = tmp \times (1 - PR^t[v_i.Id] \times \omega_{v_i,v_j}) \]

\[ \text{end if} \]

\[ \text{end for} \]

\[ PR^{t+1}[v_j.Id] = 1 - tmp \]

\[ \text{end for} \]

\[ \text{error} = \sum_{v_i \in V} PR^{t+1}[v_i.Id] - PR^t[v_i.Id] \]

\[ \text{end while} \]

\[ \text{return } PR^{t+1} \]

5 Experimental Results

This section provides experimental validation of the proposed method described in the previous section for estimating the number of infected nodes in a social network.

5.1 Simulation of a Social Network

In order to simulate a social network, individuals are modelled by agents. Each agent has a number of behaviours such as: creating content, content re-sharing, an properties such as: a set of trust values, post propagation rate \( PPR_i \) which is a property of the agent \( i \) represented by a real number in the range of \([0, 1]\) and is randomly generated using a uniform random distribution. A vector of interests, \( V_I_i = \langle I_{i1}, ..., I_{im} \rangle \), is a property of an agent \( i \) containing \( m \) real numbers between 0 and 1. Each element \( I_{ij} \) represents the degree of interest of the agent \( i \) in the topic \( 1 \leq j \leq m \).

The wall of each agent is represented by the set \( W_i = \{w_1, ..., w_l\} \) 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,
Consume($p_k, v_i, v_j$) = \[
\begin{cases}
W_i = W_i \cup \{p_k\} & \text{if } T_{pr}(v_i, v_j) = 1 \\
do nothing & \text{Otherwise}
\end{cases}
\] (13)

The trust set $T_k = \{T_{k1}, ..., T_{kn}\}$ 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 $p_k$.

5.2 Behaviours in a Social Network

Behaviour in a social network is defined as how users interact with each other. As this paper focuses on the models of information diffusion and algorithms for calculating the number of infected nodes in a network and the capability of individuals in spreading information through the network, posting creation and re-sharing are simulated.

5.3 Simulation of Posting Creation

In this simulation, a posting $p_k$ is represented by the vector of relatedness, $VT_k = \langle t_{k1}, ..., t_{km} \rangle$. Each element of this vector is generated according to the corresponding element in the vector of interest of the user $v_i$. Hence, a number $-1 < d_l < 1$ 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, $\forall t_{kl} \in VT_k$:

$$t_{kl} = \begin{cases} 
I_{il} + d_l & 0 \leq I_{il} + d_l \leq 1 \\
0 & I_{il} + d_l < 0 \\
1 & I_{il} + d_l > 1
\end{cases}$$

(14)

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

In 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.

5.4 Simulation of Re-Sharing of Postings

Once an agent is activated, each of its neighbours gets a single chance to propagate the posting according to the probability $\omega(v_s, v_i)$. 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. If activated, the trust of agent $v_i$ in the propagating agent increases, if not, it decreases.
\[
A^{ps}(v_s, v_i) = \begin{cases} 
1 & d \leq \omega^{ps}(v_s, v_i) \\
0 & Otherwise
\end{cases}
\]

5.5 Probability of Re-sharing of a Posting

For the research reported here, the function \( f \) in Equation 4 has been implemented as a simple product of these three functions. The similarity, \( \Psi(v_j, p_k) = Correl(v_j, VI, p_k, VT) \), 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:

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

5.6 Experimental Evaluation

Algorithm 1 has been validated using a simulation described in this section. In the experimental results, several topologies with varying numbers 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 [4, 11]. 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. The experiment was repeated to achieve the results shown below. Formally,

\[
Pr^{ps}(\lambda(v_i) | \lambda(v_s)) = \frac{\# of times the node v_i gets infected}{\# of Iterations = 1000}.
\]

The properties of, and results for, the simulated networks are shown in Table 1, including number of nodes, number of edges, and maximum and average number of propagations. Note that the threshold \( \phi \) for accumulated error in the Probability Propagation Algorithm was set to 0.001 in all of the experiments.

The results of statistical analysis of these simulations are shown in Table II. 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.
6 Future Work and Conclusions

This paper proposes a formal model of a class of social network and uses it to characterize the problem of measuring influence in that network. A non-deterministic diffusion model is proposed as a behavioural process useful in characterizing influence that is trust dependent.

Section 4 provided an algorithm to measure the 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. The method is validated by simulation. It is observed that the proposed method can estimate the number of activated nodes in a network given the starting point for propagation with a high level of accuracy. Clearly, further work must be undertaken using data from actual social networks such as Facebook or Twitter.

One extension required to the work undertaken in this paper is using sophisticated 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. A trust model is required to update these values according to the history of interrelationship actions; i.e., defection or cooperation.
Table 2: Correlation between simulation results and the proposed algorithm in predicting number of infected nodes using Spearman’s correlation and Pearson correlation shown as $s, p$ in table

<table>
<thead>
<tr>
<th>Dataset</th>
<th>$s, p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Epinion</td>
<td>0.98, 0.98</td>
</tr>
<tr>
<td>Facebook</td>
<td>0.98, 0.99</td>
</tr>
<tr>
<td>Scale-free</td>
<td>0.98, 0.99</td>
</tr>
<tr>
<td>Random Graph</td>
<td>1.00, 1.00</td>
</tr>
<tr>
<td>Small-World</td>
<td>0.95, 0.99</td>
</tr>
</tbody>
</table>

References