Agents of Influence in Social Networks

Amer G. Ghanem
University of Cincinnati
Cincinnati, Ohio, USA
ghanemar@mail.uc.edu

Srinivasa Vedanarayanan
University of Cincinnati
Cincinnati, Ohio, USA
vedanasn@mail.uc.edu

Ali A. Minai
University of Cincinnati
Cincinnati, Ohio, USA
Ali.Minai@uc.edu

ABSTRACT

In recent years, social networking sites and social media have become a very important part of peoples' lives, driving everything from family relationships to revolutions. In this work, we study the different patterns of interaction behavior seen in an online social network. We investigate the difference in the relative time people allocate to their friends versus that which their friends allocate to them, and propose a measure for this difference in time allocation. The distribution of this measure is used to identify classes of social agents through agglomerative hierarchical clustering. These classes are then characterized in terms of two important structural attributes: Degree distributions and clustering coefficients.

We demonstrate our approach on two large social networks obtained from Facebook. For each network we have the list of all social interactions that took place over six months. The total number of users in the two networks is 939,453 and 841,456, with 1.4 million and 8.4 million interactions, respectively. Our results show that, based the interaction behavior, there are four main classes of agents in both networks, and that they are consistent across the two networks. Furthermore, each class is characterized by a specific profile of degree distributions and clustering coefficients, which are also consistent across both networks.

We speculate that agents corresponding to the four classes play different roles in the social network. To test this, we developed an opinion propagation model where opinions are represented as m-bit strings communicated from agent to agents. An agent receiving an opinion then selectively modifies its own opinions depending on the social and informational value it places upon communications from the sending agent, its overall agreement with the sending agent, and its own propensties. Opinions are injected into the system by agents of specific classes and their spread is tracked by propagating tags. The resulting data is used to analyze the influence of agents from each class in the viral spread of ideas under various conditions. The analysis also shows what behavioral factors at the agent level have the most significant impact on the spreading of ideas.

Appears in: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2012)*, Conitzer, Winikoff, Padgham, and van der Hoek (eds.), 4-8 June 2012, Valencia, Spain.

Categories and Subject Descriptors

C.2.4 [**Distributed Systems**]: Distributed Applications

General Terms

Measurement, Performance, Human Factors

Keywords

Social Networks, Agent-Based Models, Information Propagation, Facebook

1. INTRODUCTION

The advent of social networks is a major revolution in the history of the Internet age. Today, being 'online' is the default mode for millions of people worldwide. Many sectors of society such as business, politics, advertising, gaming, etc., have embraced social networks and have sought to exploit their potential to reach the masses. Indeed, social media has allowed even individuals to set up channels of communication that rival the commercial media in their influence. The average user on Facebook has 130 friends, and people spend over 700 billion minutes per month on the site [5]. The growth of users on Google+ has occured even faster. LinkedIn's membership rocketed from 28 million users to more than 35 million users in the six months following the September 2008 onset of economic collapse [6]. The key factor that sustains such social networking sites and the businesses based on them is the propagation of information. Every user in the network is a potential point of idea injection for mass propagation. However, it is clear that not all users can achieve the same level of influence. This depends on many factors such as the users' behavior, the semantic content of the information being propagated, the structural properties of the network, etc. In this paper, we use several large datasets to develop a behavior-based classification of agents in a social network (Facebook), and to evaluate what type of agents are likely to be influential in different situations.

2. BACKGROUND

There is already a large – and growing – body of research devoted to understanding the structure and evolution of online social communities[11][17][1][12]. Some of these studies have classified members of social networks into one of three classes[1][12] based on their structural role in the network: *Singletons* are agents with zero friends; *the giant component* comprises users who are connected directly or indirectly to a large fraction of the entire network; and the *the middle region* covers small groups whose agents only interact mainly with others within the group and not with the network at large. In this study we aim to classify users into groups based on their *interactional behavior* rather than structural position. We believe that such classification will reveal valuable insights on the way users divide their social time between their friends on the network.

The issue of how ideas spread virally in a social network has been of great interest recently. This work has considered a variety of factors that determine the importance of individual agents in a network. These range from structural properties to the nature of the users and their interactions with each other. Trusov et al. [13] proposed a method for finding influential users in a network community using the number of their logins in a particular time frame as the metric for being "influential". Though intuitively appealing, this approach did not provide any unexpected insights. Similarly, Crandall et al. [3] proposed a method to show that social influence and similarity go hand-in-hand such that users tend to form new links based on similarity, which grows their sphere of social influence and, in turn, creates more similarity with users to whom they create new links.

Simpkins et al. [19] considered the role of psychological and cognitive factors in shaping the profile of idea propagation. They postulated that an idea has a 'cognitive advantage' in being retained or accepted in a particular community which is culturally circumscribed around that idea.

Kempe et al. [8] studied the problem of maximizing information spread in a social community network using recommendation or influential propagation in the form of a decreasing cascade model. In this model, a behavior spreads in a cascading fashion according to a probabilistic rule, beginning with a set of initially "active" or "influential" nodes. In another paper [9], they propose an intuitive greedy algorithm to show that it is a more efficient way to find which set of individuals should be targeted in a network for maximizing the spread of influence. This work is closest to ours in terms of its objectives.

In [10], Kleinberg showed that it is easier to find short chains between points in some networks than others. He proved that networks that include individuals operating with purely local information are very adept at finding these short chains. Considering the dynamics of a network, Ghosh et al. [7] have proposed that predicting influential users depends not only on the structure of the network, but also on details of the dynamic processes occurring on it. They classify processes as conservative and non-conservative, and claim that information spread is non-conservative. They empirically define influence as the number of in-network votes a user's post generates. This influence measure, and the resulting ranking, is evaluated from the dynamics of voting on the social news aggregator Digg, which represents non-conservative information flow. They compare their predictions of different influence models with this empirical estimate of influence. The results show that non-conservative models are better able to predict influential users on Digg and the best predictor metric is found to be the normalized α-centrality.

3. OVERVIEW

This paper presents results from two studies:

Study I: Classification of Social Agents: In this study, we use several anonymized datasets of FaceBook interactions among large groups of agents over extended periods [22] to identify classes of agents based on interaction behavior. In particular, we consider the relative social attention agents devote to interacting with other agents, and identify four distinct types of behavior in this regard. These four classes are found to be robust across multiple datasets in terms of both interaction patterns and structural properties, indicating that they capture real differences betwee agents. We then train a neural network classifier to recognize these classes, and show that it can successfully classify agents in a novel dataset according to their interaction patterns. We also provde provisional interpretations of the four behavior patterns identified in this study.

Study II: Simulation of Influence Dynamics: In the second study, we implement an agent-based simulation using a randomly chosen subset of a network in our dataset. Agents in this simulation are labeled according to the class assigned to them in Study I and follow the corresponding pattern of relative social effort in their interactions with other agents. All agents in the system begin with certain "opinions", and communicate these to other agents. This influences the receiving agents to possibly modify their own opinions and to communicate them further. The dynamics of opinions originating in agents of different types are monitored and the relative influence for each type of agent is quantified. Essentially, the question addressed by this study is, "Which class of agents are, on average, the most suitable injection point for an idea that needs to be spread virally?"

Facebook Dataset			
Network	Nodes	Edges	Interactions
A:One Month (A1)	431,995	728,243	1,412,252
A:Six Months (A6)	939,453	273,215	7,483,904
A:One Year (A12)	1,164,003	4,555,524	24,373,015
B:One Month (B1)	451,092	827,068	1,974,590
B:Six Months (B6)	841,567	2,513,432	8442,451
B:One Year (B12)	969,047	3,317,531	22,092,564

Table 1: Statisctics of the Facebook Dataset used in this paper

4. STUDY I: CLASSIFICATION OF SOCIAL AGENTS

The goal of this study is to classify agents in a large social network into different classes based on their pattern of interaction with other agents. The approach followed is to represent the agents' interaction behavior in a suitable feature space, and to cluster the agents based on these features.

4.1 Datasets:

This study used two datasets - labeled A and B - representing agents on Facebook. Each dataset provides a *social graph* indicating "friend" links between agents, and several *interaction graphs*, indicating contacts between linked agents over a period of time. The statistics for the networks are given in Table 1. The data was obtained from [22] and used with permission.

4.2 Community Extraction

Given the large number of agents in the datasets, we decided to focus, in each network, on a subset of agents that could be considered to have significant participation in a functionally useful sense. To do this, we extracted *communities* of agents from the social networks using the *clique percolation method (CPM)* [14][4]. In CPM, a *k-community* is defined as the maximal chain of adjacent k-cliques. Two k-cliques are considered to be adjacent if they share *k-1* nodes. We decided to use $k = 5$ because a value of $k > 5$ produces very few communities, and a value of $k = 4$ produces too many small ones. Only agents belonging to at least one community were classified in the analysis below, though their interactions with all agents were taken into account. With this restriction, the number of agents and links analyzed were:

Dataset A1: 1051 nodes and 3859 edges
Dataset A6: 21690 nodes and 202611 edges
Dataset A12: 64879 nodes and 959927 egdes
Dataset B1: 2229 nodes and 8128 edges
Dataset B6: 30532 nodes and 202611 edges
Dataset B12: 53633 nodes and 652776 edges

4.3 The Devotion Measure

The interaction pattern between two connected agents i and j was measured through a quantity termed *relative devotion*, Δ_{ij}, defined as:

$$\Delta_{ij} = (I_{ij}/I_i) - (I_{ji}/I_j) \equiv D_{ij} - D_{ji} \quad (1)$$

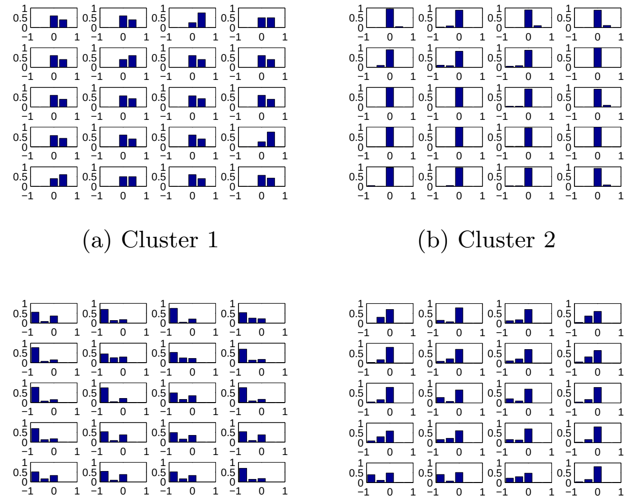
where I_{ij} is the number of interactions that i has with j, I_i is the total number of interactions for i, I_{ji} is the number of interactions that j has with i, I_j is the total number of interactions for j (note that $I_{ij} = I_{ji}$), D_{ij} is the fraction of i's interactions that are with j and D_{ji} the fraction of j's interactions that are with i. Thus, $-1 < \Delta_{ij} < 1$, where a negative value of Δ_{ij} means that i allocates a lower fraction of his/her social effort to interact with j than j is allocating to interact with i, and vice-versa for a positive value. A value of 0 means that i and j allocate the same portion of their social effort to each other.

Based on this definition, each agent i has a *devotion vector*, $\Delta_i = [\Delta_{i1}\ \Delta_{i2}\ ...\ \Delta_{in_i}]$, where n_i denotes the number of agents to which i is directly connected. Since, agents often have high-dimensional devotion vectors with variable lengths across agents, it is convenient to look at the histogram of their relative devotion values. This is obtained by distributing the relative devotion values for i into 5 bins: $[-1.0, -0.6]$, $[-0.6, -0.2]$, $[-0.2, +0.2]$, $[+0.2, +0.6]$ and $[+0.6, +1.0]$. The total is normalized to 1 and the resulting length 5 vector, $q_i = [q_i^1\ q_i^2\ q_i^3\ q_i^4\ q_i^5]$ is called the *feature vector* for agent i. This feature vector constitutes a quantitative representation of the agent in terms of its interaction behavior, and is the basis of classification.

4.4 Agent Clustering

The feature vectors obtained for indivdual agents in Dataset B6 were clustered using an agglomerative hierarchical clustering algorithm with *earth mover's distance*(EMD) as the distance measure. It is defined to be the minimum movement of probability mass required to transform one distribution into another[15]. A fast EMD algorithm developed by Pele et al.[15, 16] was used for calculations. Weighted pair group with averaging was used as the distance measure between merged clusters.

The clustering identified four types of agents in Dataset B6. Some representative feature vectors for each class are shown in Figure 1.



(a) Cluster 1 (b) Cluster 2

(c) Cluster 3 (d) Cluster 4

Figure 1: Feature vectors for 20 representative members of each class.

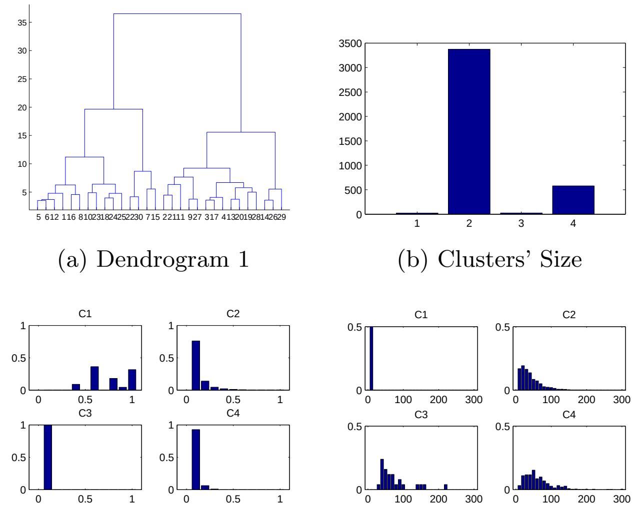
The dendogram for the clustering process is shown in Figure 2(a). Figure 2(b) shows the number of agents assigned to each class, indicating that the vast majority of agents fall into classes 2 and 4. Figures 2(c) and (d) show the distribution of the node clustering coefficient and node degree for each class. Node degree is defined as the number of links incident on the node and the clustering coefficient as the fraction of possible links that exist between the node's immediate neighbors [20][2].

It should be noted that, while only agents belonging to communities were clustered, the calculation of relative devotion, node degree and clustering coefficient used the *full network*, including unclassified agents. Thus, a significant part of the information captured in the distributions of these quantities comes from agents not included in the classification.

4.5 Neural Network Classifier

While clustering provides a useful way to organize the data, it is computationally expensive and, therefore, difficult to use with much larger datasets such as A12 or B12. Even more importantly, it does not provide a way to classify agents other than those included in the clustering process. To address these problems – and to provide further validation of the classes discovered through clustering – a neural network classifier was trained to classify agents based on their feature vectors, using the classes obtained through clustering as the true classes. The classifier had two hidden layers and was trained using the backpropagation algorithm [21, 18] on a portion of the B6 dataset and validated and tested on two other subsets of the data. Figure 3 shows the confusion matrices for the training, validation and testing case as well as over the entire B6 dataset. It is clear the the neural network was extremely successful in learning the classes.

Once trained, the neural classifier was applied to dataset B12 dataset and provided class labels for all 53,633 agents.



(a) Dendrogram 1 (b) Clusters' Size

(c) Clustering Coefficient Distribution (d) Degree Distribution

Figure 2: (a) Dendogram resulting from the clustering process; (b) Number of agents placed in each cluster; (c) Distribution of network clustering coeffiecient for each class; (d) Distribution of node degree for each class.

However, since B12 had not been processed through clustering, the accuracy of these labels could not be verified. For this, we used three methods:

Method 1: In this method, we plotted the feature vectors for 20 representative agents from each class as given by the neural network classifier (Figure 4). A comparison of these with the feature vectors given in Figure 1 shows excellent agreement, indicating that the classes assigned in B12 by the classifier were qualitatively the *same* as those found by clustering in B6. This is strong evidence that these classes are, in fact, robust across different networks, and that the classifier is able to generalize.

Method 2: Next, we plotted the degree distributions (Figure 5) and clustering coefficient distributions (Figure 6) for agents in all four classes as identified in B6 by clustering (Figures 5(a) and 6(a)) and by the neural classifier in B12 (Figures 5(b) ad 6(b)). Comparing these, it is apparent the corresponding distributions are similar, providing further evidence that the clustering and the classifier are finding the same classes in the two networks, and that each of these classes has characteristic distributions of node degree and clustering coefficients.

Method 3: Finally, we also performed a partial direct comparison by taking a subset of agents from B12, subjecting them to the clustering algorithm, and comparing the labels obtained with those given by the neural classifier. The confusion matrix for this comparison is shown in Figure 7, indicating that the two methods agreed on the classification of almost 97% of the agents.

Taken together, these results indicate three things: 1) The interaction behavior of agents in multiple social networks studied falls into four distinct and consistent classes; 2) These classes are robust and have invariant network-based characteristics across different networks; and 3) The neural network classifier is able to assign classes accurately to agents across different networks based on their feature vectors.

(a) Training (b) Validation

(c) Test (d) All

Figure 3: Confusion matrices for the neural network classifier: (a) Data used to train the network; (b) Data used to check for generalization during training but not used for training directly;)c) Data not used during training at all; (d) All data in B6.

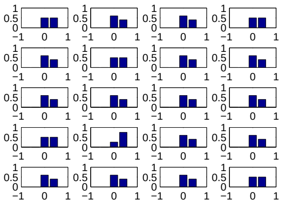
4.6 Interpretation of the Classes

A natural question that arises is whether the classes found by the above analysis are meaningful. While we are still investigating this issue in detail, some provisional suggestions can be made as follows.

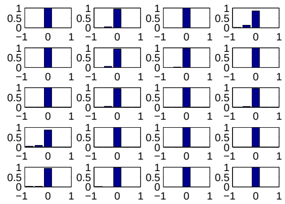
(**Class 1 - Invisibles:** This class comprises a small number of users who have very low node degree, high clustering coefficients, and are only involved in a small number of interactions, suggesting that they belong to a small, tight-knit group of friends. They mainly have positive relative devotion values, which means that their friends are more active than they are, or they are generally ignored by their friends.

Class 2 - Normals: These comprise the majority of the population, and have fairly high degree and a wide range for number of interactions. The majority of Class 2 members have low to moderate clustering coefficients, indicating that they have a broad and loosely knit group of friends, but with significant connectivity among these friends. Their directed devotion values are around 0, which means they interact with friends who interact with them.

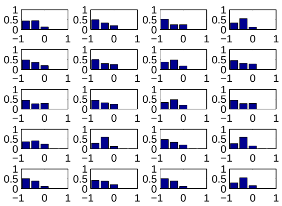
Class 3 - Celebs: This class has a small number of members with the majority being high degree nodes. The members of this class have very low clustering coefficients and a wide range of interactions, suggesting that they interact with a large, disparate set of people. Their directed devotion values are strongly negative, indicating that they allocate a smaller portion of their social effort to their friends, who actually allocate a larger portion of their social effort to them. In fact, most of the agents connected to Class 3 agents fall



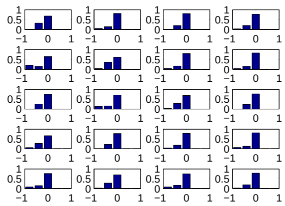
(a) Class 1



(b) Class 2

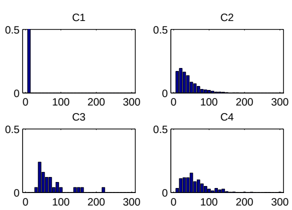


(c) Class 3

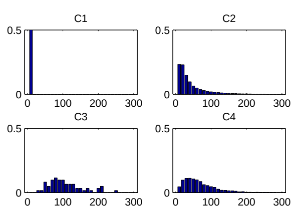


(d) Class 4

Figure 4: Feature distributions of typical members of clases assigned by the neural classifier. This figure should be comparted with Figure 1.



(a) Clustering



(b) Classifier

Figure 5: Comparison of degree distribution

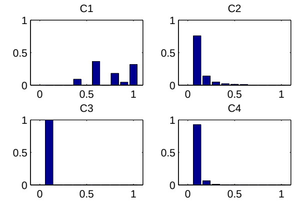
into the unclassified category and have highly positive relative devotion values (not shown).

Class 4 - Casuals: This class has a significant number of members with the majority having degrees in the range of 40-70. Most of them have low clustering coefficients, indicating that they connect with a disparate group of agents. The members of this class have a wide range of intercations and devotion values skewed slightly negative., indicating that they allocate less social effort to their friends than their friends allocate to them.

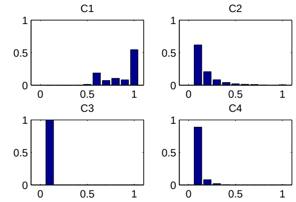
5. STUDY II: SIMULATION OF INFLUENCE DYNAMICS

A central attribute of social networks is their ability to spread information and influence – a fact used in viral marketing, political campaigning, etc. Given the very interesting agent classification described above, it is natural to ask whether agents of some class(es) are more or less influential. We addressed this issue through simulations of a multi-agent models where the spread of information injected in one agent of a particular class can be tracked across the network over time.

The propagation of influence in a network can be affected



(a) Clustering



(b) Classifier

Figure 6: Comparison of clustering coefficient distribution

	Classification ID			
Cluster ID	1	2	3	4
1	86	0	0	0
2	0	1940	0	0
3	0	0	61	28
4	0	60	0	472

Figure 7: Direct neural network classifier validation on a subset of B12 data

by a host of factors ranging from the structural characteristics of the network to the nature of the ideas being communicated. Here, we use a very simple model to study how opinions might spread in the type of network analyzed above. We consider the efficacy of agents from particular classes in spreading their opinions within a large network where the interaction behavior of individual agents follow those seen in the corresponding datasets – for both classified and unclassified agents.

5.1 Model Description

The social network used for the simulations is a subset of the A12 network described above, with about $n = 125,000$ agents. These are chosen to include a roughly proportionate sample of classified agents and all their friends, whether classified or unclassified. The numbers for each type are: Class 1 - 108; Class 2 - 40,000; Class 3 - 114; Class 4 - 4,314; Unclassified - 80,464. Each agent's type is assigned as determined by the clustering process (including the type "unclassfied"), and its interaction behavior reflects that seen in the actual dataset. Each agent, i, has a time-varying *opinion vector*, $z_i(t) = [z_{i1}(t) \ldots z_{im}(t)]$, with m elements, each indicating an opinion on some issue at time t. Each component z_{ik} can take values -1 (positive); $+1$ (negative), or 0 (don't care). The simulations used a value of $m = 7$, with all bits assigned randomnly and independently with equal probability for the three values.

During simulation, every agent i sends out one message per time-step to a randomly selected *target agent* among its friends. The probability of agent i choosing a target agent, j, at step t is given by $p_{ij}(t) = D_{ij}/\sum_{k \in N(i)} D_{ik}$, where $N(i)$ denotes the set of nodes to which i is directly connected (its circle of friends) and D_{ik} is the fraction of i's interactions that are with k according the real dataset A1, i.e., $D_{ik} = I_{ik}/I_i$. Thus, over time, the fractions of interactions i has with each of its friends reflect the proportion found in the actual dataset. A communication from agent i at time step t comprises its entire current opinion vector, $z_i(t)$. When agent j receives a message from agent i, it identifies all the mismatching components, and independently considers whether to modify its opinion on each of these to match the one received from i. The probability of doing so is determined by three *influencing factors*:

1. The *similarity*, $S_{ij}(t)$ between its own current opinion vector, $z_j(t)$, and the received opinion vector, $z_i(t)$, calculated as the number of matching bits in the two vectors. The probability of change in component k is increased by higher S_{ij}, i.e., agent j is likelier to be influenced by like-minded friends.
2. Its *relative devotion*, Δ_{ji} towards the sending agent, i. The probability of change in component k is increased by higher Δ_{ij}, i.e., agent j is likelier to be influenced by agents it holds in higher esteem (as indicated by relative devotion).
3. The *intrinsic receptiveness*, $R_j^k \in [0,\ 1]$, of agent j for component k, indicating how receptive it is to changing its opinion on this component.

The probability of changing component z_j^k to agree with z_i^k is given by:

$$p_j^k(t) = w_1 f_1(S_{ij}) + w_2 f_2(\Delta_{ij}) + f_3(R_j^k) \tag{2}$$

where the w_l are weights between 0 and 1, with $w_1 + w_2 + w_3 = 1$, and

$$f_1(S_{ij}) = \begin{cases} \frac{1}{2}[1-(1-2S_{ij})^{1-\alpha_j}] & if\ S_{ij} < 0.5 \\ 0.5 & if\ S_{ij} = 0.5 \\ \frac{1}{2}[1+(2S_{ij}-1)^{1-\alpha_j}] & if\ S_{ij} > 0 \end{cases} \tag{3}$$

where $\alpha_j \in [0, 1]$ is the *similarity influence parameter*.

$$f_2(\Delta_{ij}) = \begin{cases} \frac{1}{2}[1-(-\Delta_{ij})^{1-\beta_j}] & if\ \Delta_{ij} < 0 \\ 0.5 & if\ \Delta_{ij} = 0 \\ \frac{1}{2}[1+\Delta_{ij}^{1-\beta_j}] & if\ \Delta_{ij} > 0 \end{cases} \tag{4}$$

where $\beta_j \in [0, 1]$ is it devotion influence parameter.

$$f_3(R_j^k) = R_j^k = \gamma_j\ \forall k \tag{5}$$

where $\gamma_j \in [0, 1]$ is the *receptiveness influence parameter*.

Thus, the vector (α, β_j, γ_j) determine the *personality profile* of agent j in terms of its "influenceability", with each parameter controlling the power of a single factor as follows:

- If $\alpha_j = 0$, $f_1(S_{ij}) = S_{ij}$, i.e., linear dependence on similarity. As α_j increases towards 1, $f_1(S_{ij})$ approaches a threshold function at $S_{ij} = 0.5$, so friends with $S_{ij} > 0.5$ have very high influence on i and friends with $S_{ij} < 0.5$ have almost none.
- If $\beta_j = 0$, $f_2(\Delta_{ij}) = (1+\Delta_{ij})/2$, i.e., linear dependence on devotion. As β_j increases towards 1, $f_2(\Delta_{ij})$ approaches a threshold function at $S\Delta_{ij} = 0$, so friends with $\Delta_{ij} > 0$ have very high influence on i and friends with $\Delta_{ij} < 0$ have almost none.

Classes of agents	**(1, 0, 0)**	**(0, 1 0)**	**(0, 0, 1)**
1	0.0008 (±0)	0.0008(±0)	0.0008 (±0)
2	19.1956 (±1.2)	61.8068 (±1.79)	83.9710 (±3.01)
3	18.6787 (±0.09)	48.2217 (±1.05)	61.6270 (±1.01)
4	18.2366 (±0.27)	48.8788 (±1.08)	63.0843 (±1.01)

Table 2: Spread statistics when only one influence factor is operative

- γ_j just represents the agent's intrinsic probability to change a mismatched bit. A low value of $\gamma_j < 0.5$ indicates an agent that does not easily change its mind, and a value higher than 0.5 an agent that is easier to influence.

While exploring networks with different types of agents, etc., is potentially a rich topic for research, in the present simulations, we set $\alpha_j = \beta_j = \gamma_j = 0.5\ \forall j$, and vary the relative weights, w_1, w_2 and w_3 for each influencer. In particular, we consider the following cases:

1. **Case 1 - One Influencer:** Here, one of the weights set to 1 and the other two to 0. This measures the effect of each factor's pure influence.
2. **Case 2 - Two Influencers:** In this case, two of the weights are set to 0.5 and the third to 0. Thus, the agent is equally influenced by two factors.
3. **Case 3 - Three Influencers:** Here, all weights are set to 1/3, so all factors have equal influence.

In each simulation run, a particlar agent of a specific type is chosen as the source and two of its opinion components are tagged with a color. As these bits influence other agents to change their opinions on these two components, those bits are also tagged. At the end of the simulation, any agent with one or two tagged bits is considered *influenced*. The percentage of agents influenced over a 150 step simulation is returned as the metric of influence. For each of the cases discussed below, 20 independent simulation runs were done with each type of agent as the source, so each case involved 80 separate runs.

5.2 Results

The results of our simulations for all three cases are discussed below.

Case 1: In the first set of simulations, one of the influence weights was set to 1 individually, with the others set to 0. The results are given in Table 2. Each entry represents the perceptage of the agents in the network reached after 150 time steps, with the values in parentheses indicating the standard deviation across the 20 trials.

Case 2: In the next set of simulations, two weights at a time were set to 0.5 and the third to 0, thus exploring the joint effect of two influencers at a time. The results are given in Table 3.

Classes of agents	(0, 0.5, 0.5)	(0.5, 0, 0.5)	(0.5, 0.5, 0)
1	0.0008 (±0)	0.0008(±0)	0.0008 (±0)
2	72.0206 (±2.04)	50.7890 (±1.07)	47.0286 (±2.01)
3	54.9748 (±1.13)	47.0176 (±1.27)	42.2600 (±1.36)
4	55.7595 (±1.18)	47.7735 (±1.12)	42.6811 (±1.42)

Table 3: Spread statistics when two influence factors are operative

Classes of agents	(1/3, 1/3, 1/3)
1	0.0008 (±0)
2	69.2181 (±2.11)
3	45.9241 (±1.91)
4	47.2163 (±1.01)

Table 4: Spread statistics when all influence factors are operative

Case 3: In these simulations, all weights were set to 1/3, thus giving each influence factor equal effect. The results were as shown in Table 4:

Several conclusions can be drawn from the results given above. First, it should be remembered that the efficacy of injecting opinions in one type of the agent or another depends on several things: 1) The inherent reach of the agent, i.e., how far can the agent itself disseminate opinions through direct connections; 2) The types of agents that it connects to, and their reach patterns; and 3) The chains of devotion linking agents along the tree of connections rooted at the injected agent. We currently do not have a detailed analysis of these factors, but the results given above reflect their combined influence.

First, we consider the effect of the influencing factors, f_1, f_2 and f_3, as determined by their weights. The results show that:

- Influence spreads least when the first factor – f_1, similarity of opinion – dominates. Given the model used, the mean initial similarity among any two agents is about 0.11, giving an influence factor $f_1 = 0.0584$. Thus when the weights are set as (1, 0, 0) (first table, column 1), any mismatched bit in a receiving agent has only a 5% chance of being influenced. It is also clear that, in this situation, the class of the injected agent has virtually no effect.

- Influence spreads most when the third factor – f_3, inherent receptiveness – dominates. This is because f_3 is set to a high value (0.5) for all agents, so a receiving agent has a 50-50 chance of changing a mismatched bit. In this case, the class of the injected agent does matter – probably due to differences in connectivity and clustering.

- When relative devotion is the only factor affecting influence (first table, column 2), the spread is between that seen in the other two cases, reflecting variation in relative devotion across the network. Here too, the type of agent matters, which is expected given the different distributions of relative devotion for each class.

Next, we consider the differences among agent types in terms of their efficacy as points of injection. The following effects are observed:

- Type 1 agents are of virtually no use in propagating opinions. This reflects their low connectivity, high clustering coefficients, and, bove all, their mostly positive relative devotion values – prcluding them for being influential even with the friends they do reach.

- On average, Type 2 agents are the most effective injection points for opinions. They have high connectivity and low clusterng coefficients, indicating the ability to spread opinions to many different parts of the network. With relative devotion values near zero, they also have significant influence (f_2 near 0.5) over their large circle of friends. Interestingly, the advantage of Type 2 agents is affected most strongly by the strength of the similarity factor (f_1) in the process. We speculate that this reflects the fact that most friends for Type 2 agents have devotion values near zero towards them, while friends of Type 3 and 4 agents have mostly positive values (as implied by the mostly negative devotion values of Type 3 and 4 agents themselves). Thus, Type 2 agents rely mainly on their reach and the inherent influenceability of their friends rather than devotion to convince others. When these factors are reduced, Type 2 agents lose almost all of their extra influence, while Type 3 and 4 agents can still rely on devition (factor f_2) to compensate somewhat.

- Type 3 and Type 4 agents have almost the same level of influence in all situations. Both classes have high degree and low clustering coefficients, so they can spread quite broadly. Additionally, their own mainly negative Δ values indicate their most of their friends have positive relative devotion towards them, so when $w_2 > 0$, their influence is boosted by this.

A full analysis of the effects of influencing factors and agent types requires a much more detailed investigation, both through more simulation cases and through analysis of network connectivity patterns. Results from such investigations will be reported in the future.

Acknowledgment This work was supported in part by a National Science Foundation CreativeIT grant to Ali Minai (IIS-0855714). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

6. REFERENCES

[1] L. Backstrom, D. Huttenlocher, and J. Kleinberg. Group Formation in Large Social Networks : Membership , Growth , and Evolution. *Science*, 2006.

[2] A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–511, 1999.

[3] D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri. Feedback effects between similarity and social influence in online communities. In *In Proceedings of the 14th Annual ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2008.

[4] I. Derényi, G. Palla, and T. Vicsek. Clique Percolation in Random Networks. *Physical Review Letters*, 94(16):3–6, Apr. 2005.

[5] Facebook. Facebook statistics, Jan. 2011.

[6] M. Fraser and S. Dutta. The business advantages of social networking. *Finance and Management, the month magazine of the ICAEW's Finance and Management Faculty*, 168, 2009.

[7] R. Ghosh and K. Lerman. Predicting influential users in online social networks. In *In Proceedings of KDD workshop on Social Network Analysis (SNAKDD*, 2010.

[8] D. Kempe, J. Kleinberg, and . Tardos. Influential nodes in a diffusion model for social networks. In *In Proceedings of the 32nd International Conference on Automata, Languages, and Programming*, pages 1127–1138, 2005.

[9] D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the spread of influence through a social network. In *In Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003.

[10] J. M. Kleinberg. Navigation in a small world. *Nature*, Aug. 2000.

[11] R. Kumar and A. Tomkins. Structure and Evolution of Online Social Networks. *Work*, pages 611–617, 2006.

[12] J. Leskovec, K. J. Lang, and M. W. Mahoney. Community Structure in Large Networks : Natural Cluster Sizes and the Absence of Large Well-Defined Clusters âĹŮ arXiv : 0810 . 1355v1 [cs . DS] 8 Oct 2008. 2008.

[13] T. M., A. Bodapati, and R. Bucklin. Determining influential users in internet social networks. *Journal of Marketing Research*, (643):2010.

[14] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, pages 1–10, 2005.

[15] O. Pele and M. Werman. A linear time histogram metric for improved sift matching. In *ECCV*, 2008.

[16] O. Pele and M. Werman. Fast and robust Earth Mover's Distances. *2009 IEEE 12th International Conference on Computer Vision*, pages 460–467, Sept. 2009.

[17] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi. Defining and identifying communities in networks. *Proceedings of the National Academy of Sciences of the United States of America*, 101(9):2658–63, Mar. 2004.

[18] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *Nature*, 323:533–536, 1986.

[19] B. Simpkins, W. Sieck, P. Smart, and S. Mueller. Idea propagation in social networks: The role of cognitive advantage. In *1st ITA Workshop on Network-Enabled Cognition: The Contribution of Social and Technological Networks to Human Cognition*, July 2010.

[20] D. J. Watts and S. Strogatz. Collective dynamics of "small-world" networks. *Nature*, 393:440–442, 1998.

[21] P. Werbos. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. PhD thesis, Harvard University, 1974.

[22] C. Wilson, B. Boe, A. Sala, K. P. N. Puttaswamy, and B. Y. Zhao. User interactions in social networks and their implications. In *Proceedings of the 4th ACM European conference on Computer systems*, EuroSys '09, pages 205–218, New York, NY, USA, 2009. ACM.