

# Identifying Influencers in Social Networks

**Kushal Dave**

IIIT-Hyderabad  
Hyderabad, India  
kushal.dave@research.iiit.ac.in

**Rushi Bhatt**

Yahoo! Labs  
Bangalore, India  
rushii@yahoo-inc.com

**Vasudeva Varma**

IIIT-Hyderabad  
Hyderabad, India  
vv@iiit.ac.in

## Abstract

The central idea in designing various marketing strategies for online social networks is to identify the influencers in the network. The influential individuals induce “word-of-mouth” effects in the network. These individuals are responsible for triggering long cascades of influence that convince their peers to perform a similar action (buying a product, for instance). Targeting these influentials usually leads to a vast spread of the information across the network. Hence it is important to identify such individuals in a network. One way to measure an individual’s influencing capability on its peers is by its *reach* for a certain action.

We formulate identifying the influencers in a network as a problem of predicting the average depth of cascades an individual can trigger. We first empirically identify factors that play crucial role in triggering long cascades. Based on the analysis, we build a model for predicting the cascades triggered by a user for an action. The model uses features like influencing capabilities of the user and their friends, influencing capabilities of the particular action and other user and network characteristics. Experiments show that the model effectively improves the predictions over several baselines.

## Introduction

The rapid development of social networks such as Facebook, Flickr, Twitter, Linked-In on the Internet has resulted in social influence emerging as a complex force, governing the diffusion of the influence in the network. The emergence of social influence has allowed various companies to look beyond direct marketing to find potential customers to target. The rich neighborhood information that a social network provides about a user, can be leveraged to make intelligent marketing decisions.

Viral marketing involves identifying potential customers who can leverage their social contacts to influence their friends to perform certain action (such as clicking an ad). One way to quantify the influence exerted by these individuals is to predict the average length of cascade they trigger among their friends for a certain action. Once these individuals are identified they can be targeted to achieve large cascades and hence a wide reach. We try to tackle this problem of predicting average cascades triggered by an individual for a given action using a machine learning approach.

Copyright © 2011, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Most of the existing research work quantifies the influence as a function of the influencing capabilities of the target user only. Ideally, it should also depend on how susceptible a friend is to getting influenced by the target user. Indeed, a friend with lesser susceptibility to influence may ‘minify’ some of the influence from the user. Besides, the depth of the cascade also depends on the particular action to be propagated in the network. For example, a target user may influence a friend to click on an ad on buying a football match ticket, but she may not be able to convince to click on an ad on iPhone. Aral et al. (2010) talk about how incorporating viral features in the product (action) can induce peer influence in the network. This phenomenon of triggering of a cascade for an action as a function of the target user, their friends and the action itself is well explained by Watts and Dodds (2007) as -“The triggering of cascades in a network has numerous analogs in natural systems. For e.g, some forest fires are many times larger than average; yet no one would claim that the size of a forest fire can be in any way attributed to the exceptional properties of the spark that ignited it or the size of the tree that was the first to burn. Major forest fires require a conspiracy of wind, temperature, low humidity, and combustible fuel that extends over large tracts of land”.

From the above discussions it is clear that the concept of an influencer varies based on the particular action performed by that user. Tang et al. () show that users have varying degree of influences for different topics. Thus, the problem of identification of influencers should also take the particular action into account while identifying the influencers.

Based on these foundations, we first analyze how factors like user’s and their neighborhood’s influencing ability and action popularity affect a cascade at a user for an action. Next, we build a regression model that predicts the average cascade triggered for an action by a user. The model uses features like influence capabilities of the target user and his/her friends, how prone the friends are to getting influenced by the target user, the influencing capability of the action, and other network and user characteristics.

In a nutshell, the major contributions of the paper are:

- We empirically try to find if, apart from the user’s influence abilities, other peripheral factors like the popularity of the action, influence abilities of the user’s neighborhood also play a role in the spread of the contagion.

- We propose a novel method to identify individuals who can lead to large cascades of information in a social network using a predictive model.
- We use a social graph generated from Flickr for our experiments. The data has about 1M users, about 1B action events and some 200K odd distinct actions.

### Related work

The work most relevant to our proposed method is by Richardson and Domingos (2001; 2002), Kempe et al. (2003), Hartline et al. (2008) and Leskovec et al. (2007). Richardson et al. (2001) present a probabilistic model to mine the network value of each individual based on the influence exerted by the user on her neighbors. They show that user's network value and her intrinsic value can be combined to make optimized marketing decisions. Kempe et al. (2003) propose a greedy hill climbing strategy for picking top- $k$  influentials which works better than certain network heuristics like *degree-centrality*. Chen et al. () improve the running time of the greedy algorithm and propose certain degree-discount heuristics that improve the influence reach. More recently, Bakshy et al. (2011), analyze the propagation of influence in twitter and explore various marketing strategies governed by the cost of identifying the influencers.

The problem of identifying top influentials has been modeled in various other ways. Leskovec et al. (2007) formulate it as a problem of outbreak detection, while, Hartline et al. (2008) pose it as a problem of revenue maximization. In most of the previous work, the analysis of influence propagation was confined to the users and their social neighborhood. In this work, we try to find if the particular action that propagates also plays a role in the propagation.

Recently, there has been some work on estimating the influence probabilities (probability of a user influencing others). Tang et al. () argue that the influence exerted by an individual varies across topics. Goyal et al. (2010) outline various static and dynamic models for estimating the influence probabilities. Saito et al. (2008) employ expectation-maximization (EM) algorithm to learn the influence probabilities for the independent cascade model.

Singla et al. (2008) go on to show that people who are connected often share their interests and personal characteristics, which proves the existence of *homophily* in social networks. Anagnostopoulos et al. (2008) outline a timestamp shuffling test to assess if the social network exhibits a significant influence effect. Concurrently, Aral et. al (2009) come up with a dynamic matched sampling estimation framework that identifies both homophily and influence effects in a social network. Fond and Neville (2010) propose that influence effects are consequence of change in user attributes and homophily is present in the network if the network structure change over time. Cha et al. (2008) study the information dissemination in social graphs generated from Flickr data. Bhatt et al. (2010) build a model to predict the future adoption of the PC to Phone product for the Yahoo! IM network.

### Problem Formulation

In this section, we present the problem formulation and introduce certain terminologies. Consider a set of users in a

social network, connected with some relation  $R$ . The notion of  $R$  varies across contexts, say, in social networks such as Flickr, Facebook, Linked-In or Twitter, the relation can be - being a friend/follower, while in an Instant messaging environment relation can be interaction between the users. The relation  $R$  can be represented by an undirected graph  $(U, E)$ , where  $U$  is the set of users and  $E$  is the set of edges:  $\exists_{i,j}(u_i, u_j)$ , where  $(u_i, u_j)$  exists if and only if  $u_i, u_j$  are connected by relation  $R$ . In addition, each user  $u$  has features such as age, gender, no. of actions performed etc. For each user  $u \in U$ , we represent the set of features as  $X^u = \{x_1^u, x_2^u, \dots, x_n^u\}$ . Each user  $u$  performs certain action  $a \in A$  at time  $t_u^a$ . Action definition may vary from context to context, for example clicking on an ad, buying a product online etc. In this paper, we consider action as 'joining a group' on Flickr. With each action  $a \in A$ , we have a set of features  $S^u = \{s_1^a, s_2^a, \dots, s_m^a\}$ . Next we define the notion of action propagation.

**Action Propagation** ( $u_i \xrightarrow{a} u_j$ ): An action  $a$  is said to propagate from user  $u_i$  to user  $u_j$ , if following holds: (1).  $u_i$  and  $u_j$  are connected with relation  $R$ , that is,  $(u_i, u_j) \in E$ . (2) User  $u_i$  performs action  $a$  before user  $u_j$ , that is,  $t_{u_i}^a < t_{u_j}^a$ . (3) Action  $a$  from user  $u_j$  should follow within a certain time interval after  $u_i$  performs the action, that is,  $(t_{u_i}^a - t_{u_j}^a) < \tau$ .

The time constraint for action propagation ( $(t_{u_i}^a - t_{u_j}^a) < \tau$ ), is kept in order to have a tighter bound on the credit given to user  $u_i$  for propagating action  $a$  to user  $u_j$ . This follows inline with the findings of Anagnostopoulos et al. (2008) who avail the evidence of temporal clustering to corroborate the claims of peer influence. The significance of  $\tau$  can be explained by the fact that after performing action  $a$  at time  $t_{u_i}^a$ , if user  $u_i$  can not propagate the action to  $u_j$  (that is, make user  $u_j$  perform action  $a$ ), within time  $\tau$ , it becomes non-contagious after  $\tau$  with respect to action  $a$ . After time  $\tau$  even if the  $u_j$  performs action  $a$ , it is not credited to user  $u_i$ .

Each user  $u$  after performing a particular action  $a$ , becomes contagious for time  $\tau$  and tries to propagate the action to its neighbors in the social graph. Say, after time  $t_u^a < \tau$ ,  $u$  succeeds in propagating the action to some of its neighbors, these neighbors in turn become contagious and try to propagate the action to their neighbors and so on. This leads to a chain of actions (referred to as *cascades*), initiated at user  $u$  for a particular action  $a$ , propagating across its neighbors within few hops. These action propagations from one user to other can be well represented in the form of directed acyclic graphs. In this paper, we refer to such graphs as action graphs. **Action Graph**: An action graph for action  $a$ ,  $G_a = (U_a, E_a)$  consists of a set of Users  $U_a$  who have performed action  $a$  at some point of time and set of directed edges  $(u_i, u_j)$  such that action  $a$  was propagated from node  $u_i$  to  $u_j$ . **Propagation set**  $\vec{P}^a(u)$ : Each user  $u$  has a propagation set  $\vec{P}$  consisting of all the immediate neighbors  $u_i$  of  $u$  in social graph such that there was an action propagation from  $u$  to  $u_i$ . Formally,  $\vec{P}^a(u) = \{u_i | u \xrightarrow{a} u_i\}$

Sample graphs for an action  $a$  initiated at users  $u_1$  and  $u_7$  are shown in the Figure 1. Identifying the set of users who can trigger large cascades for a particular action is of great

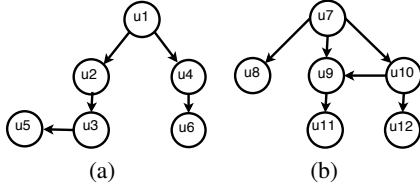


Figure 1: Sample action graphs for an action  $a$

interest in various contexts. For example, an advertiser might want to start showing a particular ad to these special set of identified individuals who can promise more reach through her neighbors. The reach of a user can be quantified by the average number of cascades triggered by a user for an action  $a$ , which leads us to the definition of *reach*. Let  $reach^a(u)$  is the reach of a user  $u$  for an action  $a$ , and can be recursively defined as follows:

$$reach^a(u) = \begin{cases} \sum_{u_i \in \vec{P}^a(u)} 1 + \frac{1}{2} * \sum_{u_j \in \vec{P}^a(u_i)} reach^a(u_j) & \text{if } \vec{P}^a(u) \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

A user gets a credit of 1 for an action propagation to its immediate neighbor. The significance of  $(\frac{1}{2})$  times the reach of the descendants of a user  $u$  in the action graph can be understood as assigning decaying credit as we move farther from the node  $u$  in the action graph<sup>1</sup>. For example, in Figure 1(a),  $u_1$  gets the complete credit for the propagation of action to  $u_2$  and  $u_4$ , while gets credit of 0.5 times  $(\frac{1}{2})$  the reach of  $u_2$  and  $u_4$ , and 0.25  $(\frac{1}{2^2})$  times the reach of  $u_3$  and so on. User  $u_1$  gets a credit of 1 for all the immediate neighbors in the action graph (i.e all the members of  $\vec{P}^a(u_1)$ ). So, the overall reach of user  $u$  for action  $a$  is  $= (1 + 0.5(1 + (0.5 * 1)) + 1 + 0.5(1)) = 3.25$ . While counting the reach of a node, we only consider reach of the descendant nodes only once, if that node is encountered again through some other path we discount it. For example, in Figure 1(b), while computing  $reach^a(u_7)$ , the edge  $u_{10} \xrightarrow{a} u_9$  will not be counted, while reach of  $u_9$  will be counted only once as  $u_9$  has already been considered through edge  $u_7 \xrightarrow{a} u_9$ . The overall reach of user  $u_7$  is  $= (1 + 1 + 0.5(1) + 1 + 0.5(1)) = 4$ . While calculating the reach, the paths are considered based on the timestamp at which they were added to the action graph. The paths are traversed in the ascending order of timestamp. In Figure 1(b), the descendants of  $u_7$  were added to the action graph in the following order:  $u_8, u_9, u_{10}$ . Hence these nodes will be traversed in the same order.

Identifying the small set of users who can elicit greater reach for an action can be formulated as the problem of predicting the reach for each user and for a particular action. Now we can formally define the problem as:

**Problem:** Given a social graph and past action events, accurately predict the reach of each user for a particular action ( $reach^a(u)$ ).

<sup>1</sup> Any appropriate value would have sufficed, we propose to use  $(\frac{1}{2^d})$  for a depth  $d$

## Dataset

To apply our framework, we use Flickr social network data for the experiments. The dataset is a longitudinal combination of the following four datasets : (1) User data ( $X$ ), which contains information about the Flickr users (2) Contacts data ( $u_i, u_j$ ): This data gives the friends information. We use this data to build the social graph. (3) User-group membership ( $u, a, t_u^a$ ): contains information about a user joining a particular group and the time of joining the group. (4) Group data ( $S$ ): tells various details about a particular group such as number of members, topics for the group.

### Algorithm 1 Computing the Action Graph

---

```

1: Input:  $C: (u_i, u_j), A: (u, a, t_u^a)$ 
2: Output:  $G_a = (U_a, E_a)$ 
3: for each tuple  $(u_i, u_j)$  in  $C$  do
4:   for each action event  $(u, a, t_u^a)$  in  $A$  do
5:     if entry for  $u_i \& u_j$  exists in  $A$  then
6:       if  $(t_{u_i}^a - t_{u_j}^a) < \tau$  then
7:         add  $u_i$  and  $u_j$  to  $U_a$ ;
8:         add a directed edge  $((u_i, u_j))$  to  $E_a$ ;
9:       else if  $(t_{u_j}^a - t_{u_i}^a) < \tau$  then
10:        add  $u_i$  and  $u_j$  to  $U_a$ ;
11:        add a directed edge  $((u_j, u_i))$  to  $E_a$ ;
12:       end if
13:     end if
14:   end for
15: end for

```

---

The social graph built from the above data contains  $O(1M)$  users and  $O(100M)$  edges. The action graph for the actions is built using data (2) and (3) as described in Algorithm 1.

The action graph without the  $\tau$  constraint contains  $O(1B)$  edges. Figure 2(a) shows the CDF for the propagated actions in the dataset. As it can be seen, the duration of propagation for some actions is even greater than 7 months. Figure 2(b) shows the frequency of the actions propagated within 2 weeks.<sup>2</sup> It should be noted that the x and y axis in Figure 2(b) are log scaled. As shown, it shows an exponential decay with time. The tail after 2 weeks till 7 months is quite long (not shown in the Figure 2(b)). The exponential decay can be attributed to the fact that when a user performs an action, her friends are more likely to adopt as they feel an urge to do the action and with time the urge may mitigate. Hence, if a user performs the same action as its peer after a substantially long time, there is a good chance that the user performed the action just because they have common interests (*Homophily*). In order to confidently attribute the action propagation to peer influence, we keep the  $\tau$  value to one week, which gives us a better bound on the influence. A similar approach of keeping a time constraint to distinguish peer influence from homophily has been used before in Goyal et al. (2010) as well. For each user  $u$ -action  $a$  pair, we compute the  $reach^a(u)$ , if  $u$  has ever performed action  $a$ .

<sup>2</sup>Figure 2(b) is rescaled to preserve data confidentiality

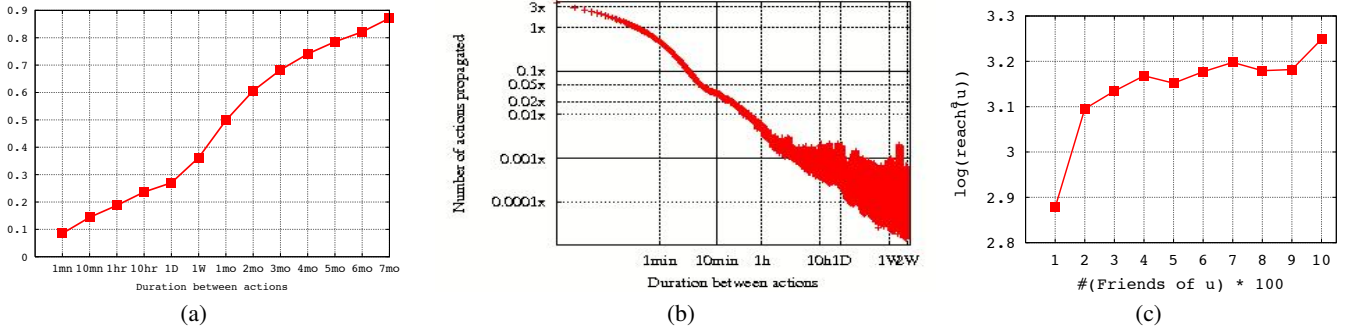


Figure 2: (a) CDF for the action events in the dataset. (b) Frequency of actions propagated within two weeks. (c) Reach v/s number of friends of  $u$

## Factors Affecting Cascades

In this section, we answer the question - what are the factors that can play a role in determining the  $reach^a(u)$  for a user  $u$  and action  $a$ . In particular, we consider various user-level, social neighborhood level and action level factors in the following subsections.

### User Level Factors

We first study, how  $reach^a(u)$  of  $u$  changes with the number of friends of  $u$  in the social network. Intuitively, more the number of friends a user has, better are her chances of propagating the action  $a$  to the next hop. Figure 2(c) shows  $reach^a(u)$  as a function of number of friends of  $u$ . The  $reach^a(u)$  values are averaged in the particular bins. For example, value 200 on x-axis represents average reach for all the users  $u$  having number of friends less than or equal to 200, but greater than 100. As shown, the  $reach^a(u)$  increases as the number of friends of  $u$  increase. This is expected as high degree users have better opportunity of propagating the action  $a$  as compared to low degree users.

Next, we analyze how  $reach^a(u)$  varies with respect to the influencing capability (*influence probability*)  $p_u^{inf}$  of  $u$ . It is the ratio of number of times an action was propagated from  $u$  to its atleast one of its immediate neighborhood by the total number of actions performed by user  $u$ .

$$p_u^{inf} = \frac{\sum_{\forall a, u_i} I(u \xrightarrow{a} u_i)}{\#\{\text{actions by } u\}}$$

where  $I$  is an indicator function taking value 1, if there was action propagation for action  $a$  by user  $u$  to atleast one of the neighbor. Ideally, more the influence probability of a user better should be its reach. As shown in Figure 3(a),  $reach^a(u)$  increases monotonically with the influence probability of the user. In addition, we refer to the extent to which a user is prone to getting influenced by others as the *prone probability* of user  $u$  ( $p_u^{prone}$ ). It is the ratio of the number of times  $u$  did an action under influence by the total number of actions performed by  $u$ .

$$p_u^{prone} = \frac{\sum_{\forall a, u_i} I(u_i \xrightarrow{a} u)}{\#\{\text{actions by } u\}}$$

The prone probability captures the susceptibility of a user to peer influence. Figure 4(a) shows the  $p_u^{prone}$  versus the  $reach^a(u)$  graph (red line). The reach increases proportionally till  $p_u^{prone}$  reaches 0.06 and after that we find a gradual improvement.

### Social Neighborhood Factors

In this section, we explore the role of a user's social neighborhood in determining her reach. We consider the influence probability  $p_{u_i}^{inf}$  and  $p_{u_i}^{prone}$  of all users  $u_i$  in the immediate neighborhood (hop one) and at second and third hop levels. The motivation behind analyzing these factors is to see if the neighborhood user  $u_i$ 's influence probability  $p_{u_i}^{inf}$ , contributes to  $reach^a(u)$ . For each user, the influence probability of  $u$ 's neighborhood at hop  $k$ ,  $p_{u:hopsk}^{inf}$ , is the average of  $p_{u_i}^{inf}$  for all users  $u_i$  at hop level  $k$  from  $u$ . Mathematically,

$$p_{u:hopsk}^{inf} = \frac{\sum_{\forall u_i: hopsk} p_{u_i}^{inf}}{\#\{u_i \text{ at hop } k \text{ from } u\}}$$

Figure 3(b), (c) and (d) plots  $reach$  as a function of  $p_{u:hopsk}^{inf}$  for  $k=1, 2$  and  $3$  respectively. As before, the  $reach^a(u)$  values are averaged in that particular bin. As shown, the neighborhood influence probabilities increase the  $reach^a(u)$  increases monotonically. As with influence probabilities, we also consider the prone probabilities of the social neighborhood up to 3 hop levels from user  $u$ .  $p_{u:hopsk}^{prone}$  is given by

$$p_{u:hopsk}^{prone} = \frac{\sum_{\forall u_i: hopsk} p_{u_i}^{prone}}{\#\{u_i \text{ at hop } k \text{ from } u\}}$$

The idea behind considering the prone probability of the neighboring users is that more susceptible the users in neighborhood to peer influence, better are the chances of the cascades increasing further. Figure 4(b),(c) and (d) shows the  $p_{u:hopsk}^{prone}$  versus  $reach^a(u)$  plot for  $k=1, 2$  and  $3$ . The sudden decline in the  $reach^a(u)$  value (for values after 0.05) can be attributed to the fact that there were very few  $p_{u_i}^{prone}$  values greater than 0.05 to have a confident estimate of  $reach^a(u)$ .

We hypothesize that if the influence probability is more than a certain threshold, we deem that factor as active and

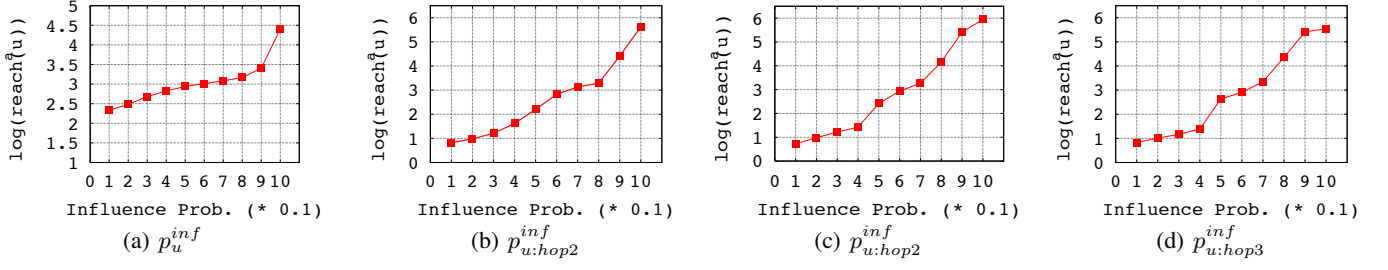


Figure 3: Effect of influence probability on the reach of user at hop level (one, two, three)

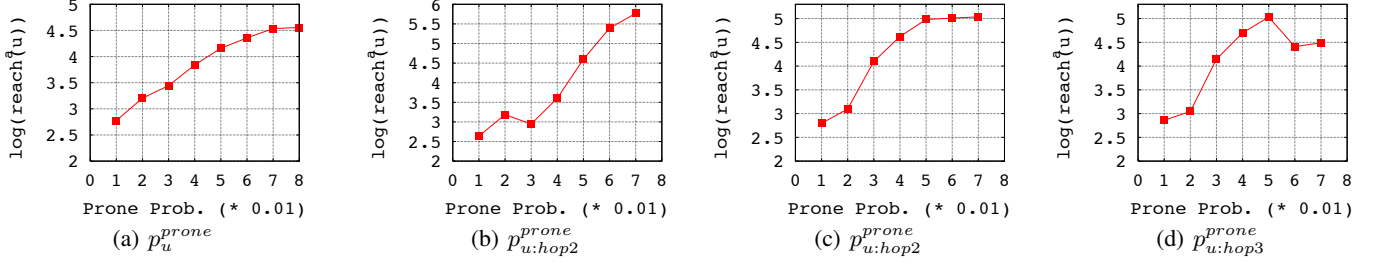


Figure 4: Effect of influence prone probability on the reach of user at hop level (one, two, three)

Active Factors	Avg. log(reach) for $p_u^{inf}$	Avg. log(reach) for $p_u^{prone}$
Only u	2.00	2.69
u+h1	2.74	3.27
u+h1+h2	3.23	3.31
u+h1+h2+h3	3.47	3.78

Table 1: Reach increases as neighborhood influence and prone probabilities cross the threshold

Active Factors	Avg. log(reach) for $p_u^{inf}$
Only u	2.21
Only a	3.60
u + a	4.04
u + a + all hops	4.24

Table 2: Reach increases as user, action and hop become active in conjunction

say that it is contagious. For example if the  $P_u^{inf}$  value is less than 0.5, we consider the user to be inactive. On the other hand, if the  $P_u^{inf}$  exceeds 0.5, the user is considered to be contagious (active). For all the user, action and neighborhood influence probabilities ( $P_u^{inf}$ ,  $P_a^{inf}$ ,  $P_{u:hopk}^{inf}$ ), the threshold is set to 0.5. Similarly, if the prone probability is less than a certain threshold, we hypothesize that the factor is not susceptible to peer influence (Inactive). In our case, if the user ( $P_u^{prone}$ ) or the neighborhood prone probability ( $P_{u:hopk}^{prone}$ ) is less than 0.03, we say that it is inactive, otherwise we consider the user/neighborhood to be active. We choose to use the average values of  $P_u^{inf}$  and  $P_u^{prone}$  as threshold. However, the resulted presented next were similar for other values of thresholds (0.3, 0.4, and 0.6) for  $P_u^{inf}$  and (0.02 and 0.04) for  $P_u^{prone}$ .

Table 1 confirms our hypothesis, where row 1 corresponds to only the user being active (both in terms of  $p_u^{inf}$  &  $p_u^{prone}$ ). Row 2 corresponds to the event that only the user and hop1 neighbors are active (while hop2 and hop3 neighbors are Inactive) and so on. As shown, as the neighborhood becomes active, the reach for user  $u$  increases (column 1). Prone probabilities (column 2) at each hop show a similar trend, the reach increases as the neighborhood at each hop becomes susceptible to peer influence.

### Action Level Factors

As with the user level and social neighborhood features, we expect the reach to increase with the popularity of the action. One way to assess the popularity, in our case, is by counting the number of users doing that action (count of users joining the group). Figure 5(a) shows the ‘count of users doing action  $a$ ’ versus the  $reach^a(u)$  plot. To find how influenceable the action is, we define action’s influencing ability ( $p_a^{inf}$ ) as the ratio of number of users doing action  $a$  under a friend’s influence by the total number of users doing action  $a$ .

$$p_a^{inf} = \frac{\sum_{\forall u_i} I(u_i \xrightarrow{a} u_j)}{\#\{\text{users } u_i \text{ doing } a\}}$$

Figure 5(b) shows  $reach^a(u)$  as a function of  $p_a^{inf}$  and confirms the intuition that as the action  $a$  becomes more influencing the reach for the action also increases. Next, we check if the action level factors combined with the user and social neighborhood factors have any impact on the reach value. As before, we fix on a threshold (0.5) and if  $p_a^{inf}$  is greater than the threshold, we say that the action is contagious (active). Table 2 analyses the impact on reach as the user, action and the neighbors become active. Row 1 gives the average reach value when only the user is active ( $p_u^{inf} \geq 0.5$  and  $p_a^{inf} < 0.5$ ). In row 2, only the action is

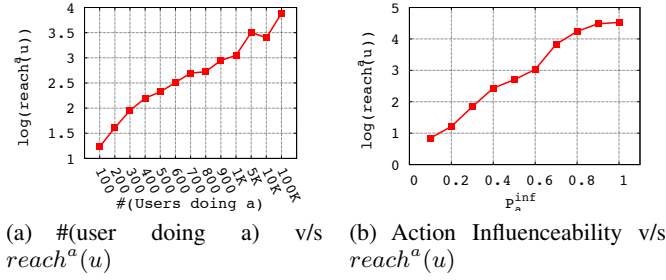


Figure 5: Action level factors

active, while in row 3, both user and action are active and so on. This shows, all the factors, when active in conjunction, can increase the  $reach^a(u)$  value further.

## Experimental evaluation

Based on our analysis in the previous section, we propose a solution to the problem posed of predicting the reach value for an action and a user. In this section, we describe the train-test split, model and features used.

### Training and Testing

From the data, we have a user-action pair and the observed  $reach^a(u)$  value. We compute the log of the  $reach^a(u)$  and use it as the reach value. For each user-action pair the goal is to predict the reach of the cascade, as if we did not know about the cascade event. Each entry in our dataset consists of the following tuple,  $(u, a, F, \log(reach^a(u)))$ . where  $F$  is the set of features described earlier.

All the feature values in  $F$  are computed on the action log built till time  $M$ . We use the data from time  $M + 1$  onwards for the experiments. The idea is to learn from the past user and action behaviors to effectively predict the reach in future for a user action pair. In our case a user only performs an action once, hence we test the model on  $(u, a)$  such that  $u$  did not perform action  $a$  earlier in time  $M$ . In cases, where a user can perform the same action more than once (for e.g. clicking on an ad), the model can also be used to predict the reach for the same user-action pair.

We split our data into ratio 60:30:10 for training, testing and validation respectively. We ensure that these sets are non-overlapping w.r.t the actions, that is, all the tuples  $(u, a, F, \log(reach^a(u)))$  having action  $a$  will go into either of the training, test or validation set. As our goal is to predict a real valued number ( $reach^a(u)$ ), we cast it as a regression problem. We use Gradient boosted decision trees (GBDT) as a regression model for predicting the reach values. The GBDT parameters, number of trees and number of leaf nodes per tree were set to 150 and 100 respectively. We use the mean square error (MSE) as our primary measure of performance. This metric is the mean squared error between the models predicted  $reach^a(u)$  value and the actual (or observed) reach value. We also use KL-divergence as the other performance measure. The improvements in the model are reported on the MSE metric. We use two baselines to compare our model:

**Baseline-1:** This baseline is the average reach of the user

$u$  across all the actions in data before time  $M$ .

$$reach^a(u) = \frac{\sum_{\forall u_i} reach^{a_i}(u)}{\#\{\text{actions by } u\}}$$

**Baseline-2:** This baseline is the average reach of the action(group)  $a$  for all the users in the data before time  $M$ .

$$reach^a(u) = \frac{\sum_{\forall u_i} reach^a(u_i)}{\#\{\text{users doing action } a\}}$$

### Features

The features used in by the model are described in Table 3. Apart from the features listed, we consider  $\log(f+1)$  as additional set of features for all numeric features  $f$ . We also had an ‘always on’ feature set to 1. In all, the model uses 41 features. In the user level feature set, apart from influence and prone probability, we also consider the average influence probability of the user across all its friends and all the actions performed.

$$avginf(u) = \frac{\#\{\text{actions propagated}\}}{n\_friends * n\_actions}$$

The social graphs and the action logs can be effectively used to measure the importance of a user in the network. Specifically, one can leverage the HITS algorithm by Kleinberg (1998) and the Page rank algorithm by Page et al. (1998) to identify the authoritative users from the graph.

The HITS algorithm gives two scores per node: Authority score and the hub score. Both these score fit well into the influencer - influenced paradigm. The authoritative score give an indication of the influencing power of a user and the hub score tries to measure susceptibility of a user to peer influence. The user rank score is similar to the page rank score, which gives the authority of the user in the action graph.

The action level features  $n\_users\_a$ ,  $n\_prone\_users\_a$  and  $p_a^{inf}$  indicate the action popularity. Besides this, we also include a Flickr specific feature topic\_cnt. Each group in Flickr mention various topics related to the group. We consider the number of topics in each group as a feature.

The social neighborhood features try to capture the capability of user  $u$ ’s neighborhood to extend the cascades triggered by  $u$ . In addition to the average influence and prone probabilities, we also compute the number of friends of  $u$  at each hop level.

## Results

In this section, we present the combined effect of various factors on the  $reach^a(u)$  value. Table 4 shows the performance of both the baselines and the machine learned model. In Table 4, Improvement 1 & 2 show improvements over baseline 1 & 2 respectively. All the results presented are statistically significant at 99% significance level. We used paired t-test for testing statistical significance.

As shown, the prediction model gives a good improvement of 48.65% over baseline-1 and an improvement of 14.15% over baseline-2. Besides, the model does better



Set	Feature	Description
User-level Features	n_friends	Number of friends of the user
	n_actions	Number of actions performed by u
	gender	M - male, F - Female, X - Unknown
	user rank	Rank of the user (similar to page rank)
	auth	Authority score of the user in the action graph using HITS algorithm
	hub	Hub score of the user in the action graph using HITS algorithm.
	$n_{uu_j}$	Number of actions propagated
	$p_u^{inf}$	Influence probability of the user
	$p_u^{prone}$	Prone probability of the user
	avginf(u)	Average influence across all users and groups
Neighborhood Features	n_hop_k	Number of users at hop level k=(1,2,3)
	$p_{u:hops}^{inf}$	Average of influence probabilities of all friends at hop k from the users
	$p_{u:hops}^{prone}$	Average of prone probabilities of all friends at hop k from the users
Action-level Features	n_user_a	Number of user who have performed action a
	n_prone_users_a	Number of users doing action a under a peer influence
	$p_a^{inf}$	Influence probability of action.
	topic_cnt	Number of topics in the group(specific to Flickr dataset)

Table 3: Feature set

System	MSE	KL Div.- (* 0.1)	Improvement1	Improvement2
Baseline-1	5.20	2.54	-	-
Baseline-2	3.11	1.53	40.19 %	-
<b>Model</b>	2.67	1.27	48.65%	14.15 %

Table 4: Improvements in the model compared to baselines

Rank	Feature	Category	Importance
1	n_users_a	action	100
2	$p_a^{inf}$	action	41.34
3	$p_u^{inf}$	user	28.63
4	n_prone_users_a	action	22.78
5	$p_{u:hops1}^{inf}$	Neighborhood	20.46
6	log(n_hop_1)	Neighborhood	14.37
7	log(n_friends)	user	12.11
8	$p_{u:hops3}^{inf}$	Neighborhood	11.65
9	$p_{u:hops3}^{prone}$	Neighborhood	11.42
10	$p_u^{prone}$	user	9.03
11	log(n_hop_2)	Neighborhood	8.79
12	log(hub)	user	8.68

Table 5: Feature importance: Top 15 features

than baseline-2 by 8.46% in terms of improvements over baseline-1.

Also, baseline-2 performs substantially better compared to baseline-1. On further investigation of the data, it was found that the average coefficient of variation for the actions was 0.29, while for users the average was 0.47. Hence lesser variance in the reach values amongst the action, results in baseline-2 performing better than baseline-1. In addition to the overall performance of the model, It is also interesting to assess the contribution of each feature in the learned model. Table 5 shows the feature importance for the top 15 features. The feature contributions are scaled with respect to the top performing feature  $n\_users$ .

As shown in Table 5, the top few performing features

$\tau$	System	MSE	KL Div. (* 0.1)	Improvement1	Improvement2
Four days	Baseline-1	4.55	2.51	-	-
	Baseline-2	2.81	1.49	38.24%	-
	<b>Model</b>	2.46	1.23	45.93%	12.45%
Six days	Baseline-1	4.84	2.58	-	-
	Baseline-2	2.93	1.48	39.46%	-
	<b>Model</b>	2.55	1.25	47.31%	12.96%
Two weeks	Baseline-1	5.77	2.60	-	-
	Baseline-2	3.32	1.57	42.46%	-
	<b>Model</b>	2.84	1.28	50.77%	14.46%

Table 6: Improvements in the model compared to baselines for various  $\tau$  values

come from the action-level category showing a healthy contribution in the overall prediction. Which means that more contagious the action better is the reach of that action for a user. Followed by the top few action level features, there are various user and social neighborhood level features showing decent contributions.

The results presented in Table 4 were for  $\tau = 1$  week. Next, we also vary the  $\tau$  value to see if changing the value affects any of the improvements obtained in Table 5. Table 6 shows the performance of the model for various  $\tau$  values. It should be noted that changing the  $\tau$  value changes the action graphs and hence the influence, prone probabilities. Most of features are recomputed for every different value of  $\tau$ . As shown in table 6 the improvements over both the baselines are consistent across various  $\tau$  values.

## Discussion

We have analyzed various factors contributing to the cascades triggered by a user. The analysis yields several interesting insights - There is a direct association between the reach and various user, action and neighborhood factors. The analysis confirms that more contagious these factors are bigger is the reach for that user and the action.

While there is an evidence of social influence, the action itself carries a large amount of predictive power augmented

by user and the neighborhood's influencing abilities. Analysis of feature contribution and the performance of baseline-2 complement the claims of action being the dominant factor in the prediction of the spread of the action. As mentioned, less variance in reach values across the actions as compared to the users results in action playing a more important role.

Bakshy et al. (2011) did a similar work of predicting the average size of the cascades for a user. Interestingly, they found out that the content itself carries little predictive power in determining the length of cascades. However, there are a few subtle differences - They focus on evaluating various targeting strategies to maximize spread of influence, while we focus on analyzing the contribution of the social network and action on the cascades. While they consider the content itself for predicting the cascades, we look at the popularity of the content as a feature. Also, as the governing social dynamics is different for both the networks, the cascades are driven by different diffusion mechanisms.

In this paper, the model learns the prediction from the past events of the user and action. In cases, where we need to identify the set of influencers for new actions for which we do not have past information, the features can be inherited from similar actions having past information. The notion of similarity largely depends upon the context. In our case, For a new group, similarity can be based on the topics that are discussed in the groups. Other intuitive example, where new actions are prevalent is the diffusion of ad's influence in social network where clicking on the ad or buying a particular product being advertised can be considered as an action. In such cases ads from the same advertiser, or for the same product can be used as a measure. In scenarios where the notion of similarity between actions can not be defined, the task of identifying the influencers has to rely on the user and the neighborhood features.

Distinguishing homophily and influence is a tough problem in general. In this paper, we avail temporal difference between the action to distinguish homophily and influence. Most of research that involves distinguishing homophily and influence is either at the network level or is difficult to implement on large online networks. There is a clear need for a more robust and scalable technique to distinguish these two types of diffusions at the action propagation granularity.

## Conclusion

In this paper, we analyzed the correlation between users, action and their reach. Analysis showed that there is a positive correlation between the reach and various user-level, action-level and neighborhood-level factors. When these factors were considered together the combined effort increases the reach value further. Based on this analysis, we built a machine learning model to predict the average reach for a user-action pair.

We empirically showed that the action, user and the neighborhood features combined together give a good prediction of the average reach of a user in the graph. While features pertaining to action play a dominant important role in the prediction, they are aptly supported by the user and neighborhood features. The model performs better than several baselines systems.

We used social graphs generated from Flickr for our experiments. It will be interesting to repeat the experiments on other social online graphs such as Twitter, Facebook or an IM network to see if they show similar trends. We consider this as a future work.

## References

- Anagnostopoulos, A.; Kumar, R.; and Mahdian, M. 2008. Influence and correlation in social networks. In *KDD '08*, 7–15. ACM.
- Aral, S., and Walker, D. 2010. Creating social contagion through viral product design: A randomized trial of peer influence in networks.
- Aral, S.; Muchnik, L.; and Sundararajan, A. 2009. Distinguishing influence-based contagion from homophily-driven diffusion in dynamic networks. *Proceedings of the National Academy of Sciences* 106(51):21544–21549.
- Bakshy, E.; Hofman, J. M.; Mason, W. A.; and Watts, D. J. 2011. Everyone's an influencer: quantifying influence on twitter. *WSDM '11*, 65–74. New York, NY, USA: ACM.
- Bhatt, R.; Chaoji, V.; and Parekh, R. 2010. Predicting product adoption in large-scale social networks. In *CIKM '10*.
- Cha, M.; Mislove, A.; Adams, B.; and Gummadi, K. P. 2008. Characterizing social cascades in flickr. In *WOSP '08: Proceedings of the first workshop on Online social networks*, 13–18. ACM.
- Chen, W.; Wang, Y.; and Yang, S. Efficient influence maximization in social networks. In *KDD '09*, 199–208. ACM.
- Domingos, P., and Richardson, M. 2001. Mining the network value of customers. In *KDD '01*, 57–66. ACM.
- Goyal, A.; Bonchi, F.; and Lakshmanan, L. V. 2010. Learning influence probabilities in social networks. In *WSDM '10*, 241–250. ACM.
- Hartline, J.; Mirrokni, V.; and Sundararajan, M. 2008. Optimal marketing strategies over social networks. In *WWW '08*, 189–198. ACM.
- Kempe, D.; Kleinberg, J.; and Tardos, E. 2003. Maximizing the spread of influence through a social network. In *KDD '03*, 137–146. ACM.
- Kleinberg, J. M. 1998. Authoritative sources in a hyperlinked environment. In *SODA '98: Proceedings of the ninth annual ACM-SIAM symposium on Discrete algorithms*, 668–677. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.
- La Fond, T., and Neville, J. 2010. Randomization tests for distinguishing social influence and homophily effects. In *WWW '10*, 601–610. ACM.
- Leskovec, J.; Krause, A.; Guestrin, C.; Faloutsos, C.; VanBriesen, J.; and Glance, N. 2007. Cost-effective outbreak detection in networks. In *KDD '07*, 420–429. ACM.
- Page, L.; Brin, S.; Motwani, R.; and Winograd, T. 1998. *The PageRank Citation Ranking: Bringing Order to the Web*.
- Richardson, M., and Domingos, P. 2002. Mining knowledge-sharing sites for viral marketing. In *KDD '02*, 61–70. ACM.



- Saito, K.; Nakano, R.; and Kimura, M. 2008. Prediction of information diffusion probabilities for independent cascade model. In *KES '08*, 67–75. Berlin: Springer-Verlag.
- Singla, P., and Richardson, M. 2008. Yes, there is a correlation: - from social networks to personal behavior on the web. In *WWW '08*, 655–664. ACM.
- Tang, J.; Sun, J.; and Wang, C. e. a. Social influence analysis in large-scale networks. In *KDD '09*, 807–816. ACM.
- Watts, D. J., and Dodds, P. S. 2007. Influentials, networks, and public opinion formation.