REMIND: Risk Estimation Mechanism for Images in Network Distribution

Dan Lin, Douglas Steiert, Joshua Morris, Anna Squicciarini, and Jianping Fan

Abstract—People constantly share their photos with others through various social media sites. With the aid of the privacy settings provided by social media sites, image owners can designate scope of sharing, e.g., close friends and acquaintances. However, even if the owner of a photo carefully sets the privacy setting to exclude a given individual who is not supposed to see the photo, the photo may still eventually reach a wider audience including those clearly undesired through unanticipated channels of disclosure, causing a privacy breach. Moreover, it is often the case that a given image involves multiple stakeholders who are also depicted in the photo. Due to various personalities, it is even more challenging to reach agreement on privacy settings for these multi-owner photos. In this work, we propose a privacy risk reminder system called REMIND, which estimates the probability that a shared photo may be seen by unwanted people - through the social graph - who are not included in the original sharing list. We tackle this problem from a novel angle by digging into the big data regarding image sharing history. Specifically, the social media providers possess a huge amount of image sharing information (e.g., what photos are shared with whom) of their users. By analyzing and modeling such rich information, we build a sophisticated probability model that efficiently aggregates image disclosure probabilities along different possible image propagation chains and loops. If the computed disclosure probability indicates high risks of privacy breach, a reminder is issued to the image owner to help revise the privacy settings (or at least inform the user about this accidental disclosure risk). The proposed REMIND system also has a nice feature of policy harmonization that helps resolve privacy differences in multi-owner photos. We have carried out a user study to validate the rationale of our proposed solutions and also conducted experimental studies to evaluate the efficiency of the proposed REMIND system.

Keywords: Image privacy, Sharing chain, Risk estimation, Probability model

1 Introduction

With social media affecting the way millions of people live their lives each day, we have assisted to an explosion of user contributed content online, especially images and media files. Some of the user-contributed photos may be harmless and effective for users' self-recognition and gratification. However, for many of these photos, the portrayed content affects individuals' social circles, as it either explicitly includes multiple users or it relates to users other than the original poster (e.g. a child or a house/location). To further complicate this issue, photos may be leaked or disclosed with an audience larger than expected, for both the image owner and its stakeholders.

To alleviate privacy concerns, many social websites provide basic privacy configurations that allow the users to specify whom they would like to share the photos with. Some social websites like Facebook offer more sophisticated privacy configuration options including the control of how the photos being re-shared among friends of friends. For example, Facebook ensures that a photo will not be re-shared through friends of friends with people who were not in the audience that the photo owner originally selected to share with. However, if a user posts a photo on another person's timeline, the photo owner will still have no control of whom will see the photo. Moreover, the current privacy settings merely block the over-sharing by removing the shared link of a photo, but do not prevent a person from downloading the photo and share it again. Figure 1 illustrates a simple example of the privacy breach caused by image sharing propagation. Figure 1(a) shows that Alice usually shares funny photos with all her friends, and the same photos are often re-shared back to one of Alice's friend (Mary) after propagation. Later, when Alice wanted to share a new funny photo of herself with friends but excluding Mary (Figure (b)), Tom and Mike who are in the network may still behave the same by attempting to re-share the photo with their friend Mary. Even if some social sites like Facebook block re-

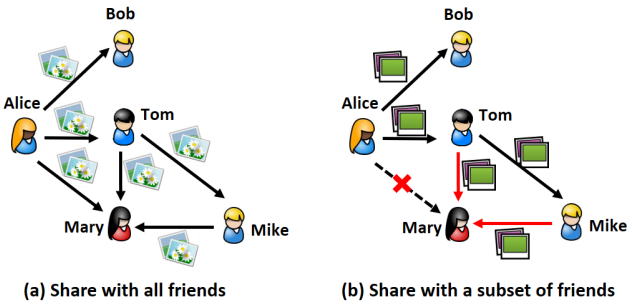


Fig. 1. **An Example of Privacy Breach Due to Image Propagation**

- *Dan Lin (corresponding author), Douglas Steiert and Joshua Morris are with EECS Department, University of Missouri, USA, e-mail: lindan@missouri.edu, djsg38@mail.missouri.edu, jdm6b3@mail.missouri.edu*
- *Anna Squicciarini is with Information Science and Technology, Pennsylvania State University, USA, asquicciarini@ist.psu.edu*
- *Jianping Fan is with CS Department at University of North Carolina at Charlotte, USA, e-mail: jfan@uncc.edu*

sharing with people who are not in the original sharing list of Alice, Tom (or Mike) would notice the failure of his direct re-sharing with Mary. There are several possible reactions. One possibility could be that Tom (or Mike) may feel that it is just a temporary technical issue of the website, and simply download the photo and re-share it with Mary again. Another possibility is that Tom may tell Mary she has been blocked by her friend about this photo. In any case, there is a privacy risk that Alice may not be aware of as she may have thought her privacy setting already fully protects her privacy.

The potential privacy risks caused by sharing from friends to friends have been aware by many [1], [2], [3], [4]. Some propose monitoring approaches to check the privacy violation during each sharing event [1], [2]. Most employ [5], [6], [7], [8], [9] clustering techniques to classify users based on their privacy preferences, profile similarities, social network topology, image content and metadata, in order to identify risky users and recommend better privacy policies. However, to the best of our knowledge, none of the existing works leverages the image sharing history and develops probability models to provide a straightforward view of the sharing consequence as we will elaborate shortly in this work.

Another critical factor that could cause a privacy breach is the difference among privacy preferences of people depicted in the same photo. Due to the variety of personalities, users may drastically disagree on the scope of sharing for a given co-owned image causing some significant conflicts. In some instances, it can be courtesy that these users may personally discuss which photos can be posted and by whom so that there are no conflicts of interest, but that takes time and is not often the route pursued. An increasing number of recent works [10] have analyzed how to address the policy conflict, by considering every user's prior privacy preferences of sharing or through semi-automated resolution mechanisms. These existing works are usually based on fuzzy logics while we aim to provide clear evidence of chances of privacy breach.

In this work, unlike any existing works, we tackle the privacy risk estimation problem from a novel angle by digging into the big data regarding image sharing history. Specifically, the social media providers possess a huge amount of image sharing information (e.g., what photos are shared with whom) of their users. In fact, even some external websites [11] have provided tools to maintain statistics of the sharing propagation throughout the social networks.

By analyzing and modeling the rich information of image sharing history, we build a sophisticated probability model that aggregates image disclosure probabilities along different possible image propagation chains and loops. We present the users with direct evidence of potential scope of sharing, i.e., the probability of unwanted people to access one's photos. These unwanted people could be the people who are in the photo owner's contact list but not in the sharing list, or the people who are clearly specified as "not-to-share" by the photo owner. If the computed disclosure probability indicates high risks of privacy breach, a reminder will be issued to the image owner to help revise the privacy settings. Users then have the opportunity to make informed decisions when setting their privacy preferences. For example, our work would remind Alice that sharing with Tom could result in 90% chance that Mary would see the photo as well. Based on such a high disclosure probability, it is very likely that Alice would remove Tom from her initial sharing list, and hence avoid the potential privacy breach. An initial user study (Section 2) shows that more than 75% of users will do so.

Carrying the spirits of our goal, the proposed system is named REMIND (Risk Estimation Mechanism for Images in Network Distribution). Our proposed system would be a great add-on to be adopted by social networking providers such as Facebook to further improve users' privacy protection. According to a study [12], only 37% of users say they have used the Facebook's privacy tools to customize their settings. That means even if the privacy tools allow users to directly control who they do not want to share or re-share, these tools still may not be used by the users if they are not aware of the potential risk of privacy breach. We envision that the percentage of the use of privacy configuration tools would be greatly improved if privacy breach reminders are shown to the users. To actually examine the effectiveness of our system, we conduct a A/B test in a simulated social network environment. The results demonstrate that users will accept privacy reminders and make changes to privacy settings when the privacy reminder matches their privacy concerns. To sum up, the REMIND system has the following novel contributions:

- The REMIND system provides users a quantitative and easier way to directly evaluate the potential consequence of sharing. It "nudges" users about the risk of sharing the image with certain people and remind the owners of the photos about users that could be explicitly excluded. The goal is to reduce the risk of privacy breach that could result from the image propagation in the social network.
- Underlying the REMIND system, we propose a sophisticated probability model that models the image sharing history. It is very challenging to calculate and aggregate the disclosure probabilities caused by various sharing paths especially loops in convoluted social networks. To overcome this, we design an efficient probability serialization algorithm that ensures each node in the related social circle to be visited and calculated only once.
- The REMIND system also has a nice feature called policy harmonization, which calculates *image disclosure matrix* to help resolve differences in the privacy preferences of people depicted in the same photo.
- We have carried out a user study in a simulated social network environment to validate the rationale of our proposed solutions and also conducted experimental studies in real-life social networks to evaluate the effectiveness and efficiency of the proposed REMIND system.

In addition, it is worth noting that while we present our models with images as a reference content type, any co-owned or co-managed piece of content in an online social setting could take advantage of REMIND, with no significant differences.

The remaining of the paper is organized as follows. Section 2 presents the results from an exploratory user study that demonstrates the potential benefits of the REMIND system. Section 3 reviews the related work. Section 4 introduces the problem statement. Section 5 elaborates the proposed REMIND system. Section 6 reports the experimental study and Section 7 concludes the paper.

2 An Exploratory User Study

This user study aims to examine the need of the REMIND system by investigating how a typical user would react if the user knows that sharing with someone may cause a privacy breach with different disclosure probabilities. The user study has received the IRB approval from the university. The research is conducted in an anonymous form which means we do not record any information about the participants that could be used to identify them. We created an online survey which asked for demographic information, photo sharing preferences, and then presented users with various sharing scenarios. In what follows, we first describe the demographics information of people who participated in the user study, and then analyze the results of the users' responses.

The user study involves 114 users who are recruited online. There are 33 females and 79 males. Their ages range from 18 to over 50. The study consists of two parts. The first part collects demographics and data about online social networking habits, as shown in Table 1. The second part collects participants' reactions regarding privacy settings when they know the probabilities of their photos being seen by unwanted people.

From the response, we see that all of the 114 users have at least one social media account with about 71% of them have more than 2 accounts. This is consistent with current data on social network usage. When asked how often they shared images on social media, more than half of the participants confirmed that they share regularly (i.e., either a few times a week or a few times a month), and 6% admitting they share images every day. Over half of the participants estimate having shared a total of 50 to over 200 images. To understand how conscious the participants were about the privacy of their images, we asked them whether they often designate a group of people that they would like to share when uploading a photo. Although about 72% of participants answered yes, we can see there is still a significant percentage (28%) of users who simply make their personal photos public. We note that there is no statistical correlation between number of accounts or frequency of sharing with users' privacy habits, confirming that users appear unwilling (or unable) to set own privacy settings regardless of the amount of content actually disclosed online. Moreover, even users who set up the privacy configurations during the photo sharing, they still may not have the knowledge what would be the final audience of their images if their friends re-share the received images. To get an idea of how much of an impact the social network connections can make on a user's privacy, we asked how many contacts each user had in their social media accounts. 53% of the participants claim to have between 100 to 500 contacts while 21% claim to have more than 500 contacts. Imagine that even half of those contacts sharing the images they see to their own additional contacts, we can see how quickly an image can spread which may result in undesired privacy breach that the photo owners do not anticipate. Here, we note that frequency of posting is negatively correlated with amount of content posted (Pearson= -.223, $p<0.5$), but again there is no statistically significant correlation between users' frequency of managing sharing settings with number of friends.

In the second part of the study, subjects were presented with three different scenarios. In the first scenario, the photos to be shared are about the photo owner doing an extreme sport that the photo owner does not want his/her close family members to worry about. In the second scenario, the photos are about the photo owner who is doing some crazy stuff in a party and only wants to share with close friends. In the last scenario, the photos show the photo owners in funny costumes which are intended to only share with family members instead of co-workers or managers at work. The first two scenarios are designed to capture the case single-owner content decision making processes, and the last scenario is about multi-owner decision making. For each scenario, we present sample photos to the participants to help them better understand the scenarios. Then, we ask whether they would consider excluding a person from their initial sharing lists if they know there is 90%, 50%, or 10% chance that the photo may be disclosed to unwanted people by that person. Table 2 reports the percentage of participants who responded positively that they would change their initial privacy settings as suggested. All responses are positively correlated with a Pearson coefficient of 0.734 (case 90% and 50%) and 0.33 (50% to 10%), and $p<0.05$. From the table, we can clearly observe an increasing trend of privacy concerns when the relationship between the photo owner and the possible viewer becomes loose. Specifically, 61% of participants said they would not share with a person if there is 90% chance that the photo may be disclosed to their close family members who are not in the initial sharing list; the percentage jumps to 78% when there is 90% chance of disclosure to the photo owner's friend who is not supposed to see the photo; the percentage further increases to 85% when it is about the disclosure to users tend to be less concerned about privacy within close social circles. The second observation is that the percentage of participants who agrees to change the privacy settings decreases with the disclosure probability. For example, when there is only 10% chance of disclosure to undesired people, only around 30% of people chose to restrict the privacy settings in the first two scenarios. There is still a high percentage (68%) of people would like to prevent their manager from seeing the photo even there is only 10% chance of disclosure.

To gain early evidence of the potential usefulness of our REMIND system, at the end of the user study, we directly asked the participants if they would like to have such kind of privacy breach reminder provided by social websites. 75% of participants responded that they are interested in using this kind of system. More specifically, majority of the people who usually set up privacy settings (72% of all the participants) are interested in receiving privacy reminders, while a small percentage of this group of people said no which is probably because they may think they have already configured their privacy settings very privately. Among the

TABLE 1
Questions about Uses of Online Social Networking

Question	Options
How many photos do you have in your social accounts?	0 to 50, 51 to 100, ..., 201 to 300, more
How many contacts do you have in your social account?	0 to 10, 11 to 50, ..., 301 to 500, more
How often do you upload photos?	Everyday, a few times a week, ..., a few times a year, rarely
Do you often designate a group of people when sharing photos?	Yes, No (I usually make my photos public)

TABLE 2
User Response to Different Scenarios

Privacy Breach Probability	90%	50%	10%
Scenario 1 (Single-owner photo, undesired disclosure to close family member)	61%	57%	34%
Scenario 2 (Single-owner photo, undesired disclosure to friend)	78%	73%	31%
Scenario 3 (Multi-owner photo, undesired disclosure to manager)	85%	80%	68%

group of the people who claimed rarely to configure privacy settings, a small percentage of this group show interests in using the REMIND system which indicates that the users started being aware of privacy issues even through this quick user study. We feel that with the REMIND system in place in the real social networks, it will gradually help enhance public awareness in privacy problems, and eventually help people gain more privacy protections.

3 Related Work

Our work shares similar goals of privacy protection with existing works on privacy policy recommendation systems, privacy risk estimation and privacy violation detection in social networks. However, our proposed probability-based approach is unique that has not been explored in the past. More details are elaborated in the following.

There have been many privacy policy recommendation systems [5], [13], [6], [7], [14], [8]. They typically utilize certain types of machine-learning algorithms to analyze users' profiles, historical privacy preferences, image content and meta data, and/or social circles, in order to predict privacy policies. Instead of relying on social circles and clustering social contexts, another thread of work looks into the image content and metadata directly [9], [15], [16], [17]. In order to even better capture the users' privacy preferences, there is a new trend of hybrid approaches which combine knowledge learned from both social contexts and the image content [18], [19]. For example, Squicciarini et al. [18] propose to utilize community practices for the cold start problem in new users and image classification based approaches for users with long privacy configuration history. Yu et al. [19] consider both content sensitiveness of the images being shared and trustworthiness of the users being granted to see the images during the fine-grained privacy settings for social image sharing.

Since our work considers the privacy breach caused by friend-to-friend sharing, we review works that also examine this aspect. Li et al. [20] present a general discussion of privacy exploits, such as leakage of employment information, through friend-to-friend sharing. Akcora et al. [1] propose a risk model that estimates the risk of adding a stranger as a new friend. They cluster users based on their profile features, privacy settings and mutual friends. Our approach is different from theirs in terms of both goals and approaches. We aim to estimate the risk of an image being seen by an unwanted person, while they aim to estimate whether a stranger could be added as a new friend. We define probability models while they use clustering techniques. Another work on malicious user identification is by Laleh et al. [2] who analyze social graphs using the assumption that malicious users show some common features on the topology of their social graphs. This work is also different from ours regarding goals and approaches. More related to our work, Kafali et al. [3] propose a privacy violation detection system called PROTOSS which checks and predicts if the users' privacy agreements may be violated due to the friends of friends sharing. Their approach is based on semantic checking and rule reasoning. The potential limitation is that the privacy violation prediction is likely to report lots of false positives in a well connected social network since the system preassumes that the sharing would happen as long as the two users are connected in the social network. In our work, our proposed probability model not only models social network topology but also the image sharing statistics to provide more refined and accurate predictions. Later, Kokciyan et al. [4] also propose a monitoring approach which utilizes agents to keep checking whether the current sharing activity (e.g., by a friend of the owner) violates the privacy requirements of the content owner. Unlike this approach that relies on agents to continuously monitor the sharing events, our approach aims to prevents the potential privacy breach at the beginning of the sharing.

Similar to our system which provides an image disclosure probability to enhance users' privacy awareness, there have also been some other types of approaches being proposed to assist individuals to make more beneficial privacy and security choices as reviewed by Acquisti et al. in [21]. Many of these approaches use privacy scores [22], [23], [24] which are defined based on image sensitivities, privacy settings and users' positions in the social network. Unlike these existing privacy scores, our work calculate the privacy risk from a different angle – the image sharing history.

Another related area of research is information diffusion. The works on information diffusion [25], [26], [27] study how information may be propagated in the social network, finding the most influential nodes (i.e., the nodes that can distribute information to a large number of nodes) or pos-

sible reactions to information sharing. Compared to these information diffusion models which have the information propagation graph as their output, our work takes the historical information propagation as the input and then calculates the privacy disclosure risk based on that.

Lastly, our policy harmonization function as an add-on feature of our REMIND system is just one way of resolving policy conflicts. We would like to mention that there have been different solutions to this policy conflict problem. For example, Hu et al. [28] formulate an access control model to capture the essence of multiparty authorization requirement and employ a voting scheme for decision making when sharing photos. Such and Criado [10] propose a set of concession rules that model how users would actually negotiate to reach the common ground. Kokciyan et al. [29] propose PriArg (Privacy for Argumentation) where agents help reach sharing consensus by negotiation. Similar to PriArg, Kekulluoglu et al. [30] propose PriNego that can follow two different negotiation strategies to help agents agree to share faster. In addition, there have also been general approaches for integrating access control policies of collaborating parties [31] which however requires the users to clearly specify how these policies should be combined.

Compared to all the existing works on image privacy preservation, our work distinguishes itself in two main aspects. First, to the best of our knowledge, it is the first time that the large volume of image sharing statistic data being considered and sophisticated probability models being built for privacy risk estimation. Second, compared to existing approaches which usually recommend policies based on relatively fuzzy logics, our system offers a direct and quantitative view of the risk of sharing so that the users could make more informed decisions regarding the image sharing. That is, we calculate, within a social network of varying size, the probability that different users will be able to view an image if one particular user decides to share it. In this way, it gives users an insight into just how vulnerable their photo is, and the probability someone they do not desire to view their photo will end up seeing it.

4 Problem Statement and Assumptions

Our work is developed based on the assumption that online social networking providers have full knowledge of their own social network graphs, users' profiles including privacy preferences, and their users' image sharing history.

We consider the image sharing problem in a finite social network as defined below.

Definition 1. **(Social Network)** A social network is defined as an undirected graph G(Ξ, R), where Ξ is the set of the users in this social network, and R is the set edges connecting pairs of users who have relationship with each other, i.e., $R = \{(u_i, u_j)\}$ where $u_i, u_j \in \Xi$.

Each user can specify a group of people in the same social network who are allowed to access the shared image. The privacy policy is formally defined as follows.

Definition 2. **(Image Privacy Policy)** An image privacy policy is in the form of $Pol = \{img, u, U^+\}$, where u is the image owner, and U^+ is the group of people who are allowed to access user u's image img.

Our work aims to compute the disclosure probability (as defined in Definition 3) that the shared image may be seen by people who are unwanted by the photo owner. There could be two types of unwanted audience. One type of such audience could be those who are in the photo owner's contact list but are not included in the photo owner's original sharing list. The other type of audience could be those who are clearly specified by the photo owner as "not-to-share", and these people may or may not be in the photo owner's contact list. For example, a photo owner may not have his boss in his contact list, but the photo owner may still include his boss in the list of "not-to-share" for certain kinds of photos as long as his boss social account name is known.

Definition 3. **(Image Disclosure Probability)** Let u_o denote the owner of an image img, and U_o denote the set of users in u_o's contact list. Let $Pol = \{img, u_o, U_o^+\}$ denote the corresponding privacy policy for image img. The image disclosure probability $P_{u_o \Rightarrow u_t}$ is the probability that user u_o's image may be seen by a target user u_t, where $u_t \in U^-$, and U^- is the set of the users who are not wanted to see the photo by u_o.

5 The REMIND System

We propose a REMIND (Risk Estimation Mechanism for Images in Network Distribution) system that presents the image owner a privacy disclosure probability value that indicates the risk of his/her image being viewed by an unwanted person. The REMIND system not only works for photos with single owners, but can also be utilized to help resolve privacy differences in multiple users depicted in the same image. Figure 2 gives an overview of the data flow in the REMIND system.

First, the REMIND will identify the list of people who the image owner u_o does not want to share image with, i.e., U_o^-, by analyzing the policies associated with the image. Note that for an image with multiple users, e.g., u_{o_1}, u_{o_2}, ..., u_{o_n}, this step will return a set of $U_{o_i}^-$ whereby the $U_{o_i}^-$ is the list of people that user u_{o_i} does not want to share the photos with. The second step is to conduct the risk analysis for each user in U_o^-. We will first extract the sub-network connected to the owner(s) of the photo and then calculate the image disclosure probability for the image owner(s) with respect to the users (u_t) in U_o^-. If the computed disclosure probability $P_{u_o \Rightarrow u_t}$ is above certain threshold (e.g., 80%) defined by the photo owner, the REMIND system will issue an alert to the image owner u_o regarding this. The alert will clearly indicate through which user who are in the original sharing list, user u_t may have the chance of $P_{u_o \Rightarrow u_t}$ to view the shared image. If the photo has multiple users in it, the REMIND system will conduct a policy harmonization process which combines all the alerts and suggests a possibly smaller group of users to share in to avoid undesired image disclosure. In what follows, we will elaborate the detailed algorithm for each step.

It is worth noting that the disclosure probability of an image is calculated with respect to the historical sharing

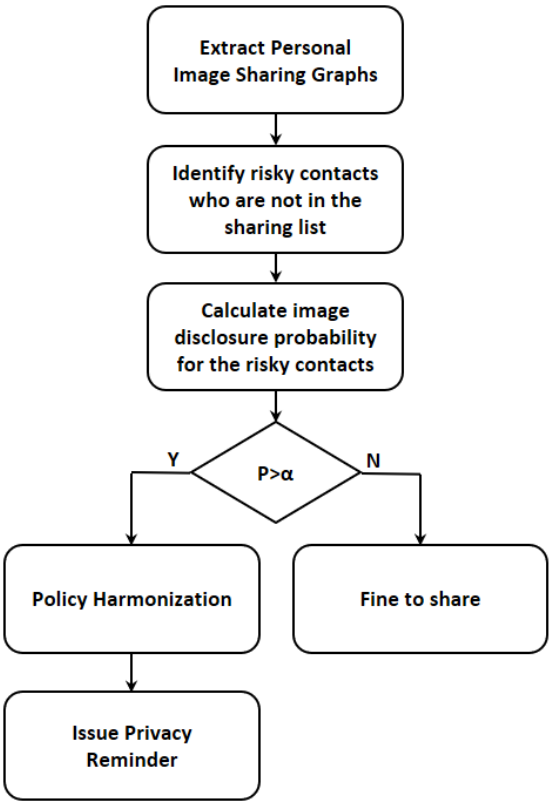


Fig. 2. **An Overview of REMIND System**

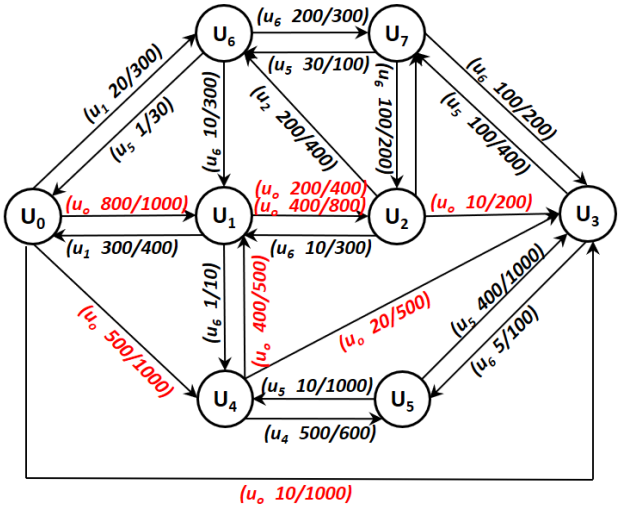


Fig. 3. **An Example of Image Sharing Graph**

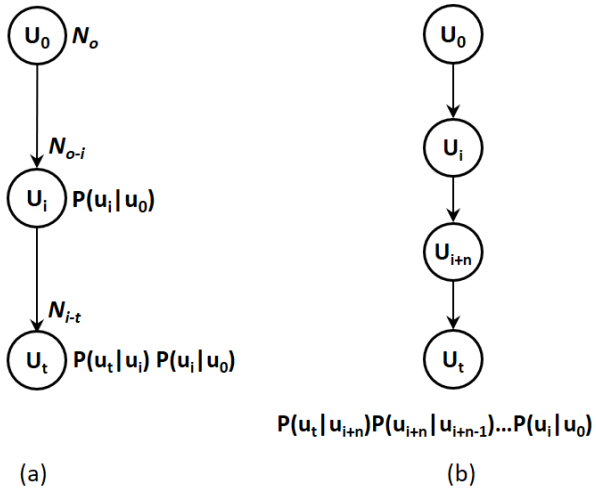


Fig. 4. **Single Photo Propagation Chains**

information of the same category of images. This is because different types of images may have different levels of privacy concerns. For example, photos which are categorized as "funny" are more likely to propagate throughout a much larger portion of the social network than photos which are categorized as "normal daily life". To obtain categories of images, images can be easily classified based on their content using existing image classification tools [9], [17]. For the ease of illustration, the subsequent calculations and examples are referring to the images of the same category.

5.1 Propagation Chain Model

As aforementioned, our goal is to calculate the probability that the photo owner's contact who is not in the original sharing list may view the shared photo via friend-to-friend sharing chains. We model such sharing propagation as an image sharing graph as follows.

Definition 4. **(Image Sharing Graph)** An image sharing graph is a directed graph SG(Ξ, SR, Ψ), where Ξ is the set of users in the social network, and SR is the set of ordered pairs of users $SR = \{\langle u_i, u_j \rangle\}$ which indicates that user u_i shares some images with user u_j, Ψ is the set of detailed image sharing information including the origin of the image and the number of shares received. Specifically, $\Psi = \{\psi_{u_o:u_i \to u_j}\}$ where $\psi_{u_o:u_i \to u_j}$ denote the number of images originally owned by u_o and are shared by user u_i with u_j.

Figure 3 illustrates a portion of the image sharing graph in a large social network. Let us take user u_o's photo sharing propagation as an example (highlighted red in the figure). Assume that user u_o has 1000 photos of her own. She shares 800 out of 1000 with her contact u_1, denoted as "u_o 800/1000" on the edge from u_o to u_1. User u_o also shares 500 her own photos with user u_4 who forwards 20 of the received photos to u_3 and 400 to u_1. Now user u_1 has u_o's photos from two sources. It is possible that u_1 shares 400 photos out of the 800 shares that she directly received from u_o with u_2, and another 200 photos out of the shares that she received from u_4 with u_2 too. Correspondingly, we see two pieces of sharing information on the arrow from u_1 to u_2. Next, u_2 further shares 10 of u_o's photos from those sent by u_1 with u_3. In addition, u_o also shares 10 out of 1000 photos directly with u_3.

Based on the image sharing graph, we proceed to discuss how to compute the image disclosure probability $P_{u_o \Rightarrow u_t}$, i.e., the probability that user u_o's photo may be viewed by user u_t through the sharing propagation chains. Let us start from the simplest case (Figure 4(a)) when there is only one intermediate user connecting the photo owner u_o and the

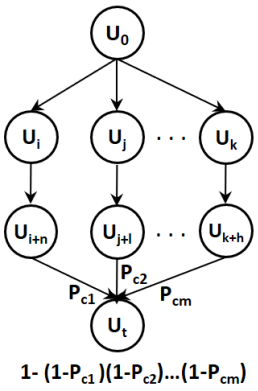


Fig. 5. **A Generic Photo Propagation Model**

target user u_t. The probability $P_{u_o \Rightarrow u_t}$ can be computed by Equation 1.

$$P_{u_o \Rightarrow u_t} = P_{u_o \Rightarrow u_i} \cdot P_{u_o}(u_t|u_i) \quad (1)$$

In Equation 1, $P_{u_o \Rightarrow u_i}$ is the probability that u_o may share photos with u_i which can also be denoted as $P(u_i|u_o)$, and $P(u_t|u_i)$ is the probability that u_t may receive u_o's photos from u_i when u_i has u_o's photos. Specifically, let N_o denote the original number of photos that user u_o possess, let N_{o-i} denote the number of photos that user u_o shares with u_i, and let N_{i-t} denotes the number of u_o's photos that u_i further shares with u_t. $P_{u_o \Rightarrow u_i}$ can be easily computed by $\frac{N_{o-i}}{N_o}$, and $P_{u_o}(u_t|u_i)$ can be computed by $\frac{N_{i-t}}{N_{o-i}}$. Then, we have the following:

$$P_{u_o \Rightarrow u_t} = \frac{N_{o-i}}{N_o} \cdot \frac{N_{i-t}}{N_{o-i}} = \frac{N_{i-t}}{N_o}$$

Next, we extend the above case to the scenario when there are multiple users in a single chain as shown in Figure 4(b). The probability that u_o's photos may reach the target user u_t via multiple users (sharing routes) of sharing can be computed by Equation 2.

$$P_{u_o \Rightarrow u_t} = P_{u_o \Rightarrow u_i} \cdot P_{u_o}(u_t|u_{i+n}) \prod_{j=1}^{n} P_{u_o}(u_{i+j}|u_{i+j-1}) \quad (2)$$

At the end, we extend the probability formula to the generic scenarios (as shown in Figure 5) when there are multiple propagation chains between the photo owner u_o and the target user u_t. The final probability $P_{u_o \Rightarrow u_t}$ is given by Equation 3, where P_{c_k} denotes the sharing probability from the chain containing u_t's direct parent u_k, and m denotes the total number of sharing propagation routes.

$$\begin{aligned} P_{u_o \Rightarrow u_t} &= 1 - \prod_{k=1}^{m} (1 - P_{c_k}) \\ &= 1 - \prod_{k=1}^{m} (1 - P_{u_o \Rightarrow u_k} \cdot P(u_t|u_k) \cdot \alpha) \end{aligned} \quad (3)$$

In Equation 3, the image disclosure probability $P_{u_o \Rightarrow u_t}$ is computed by aggregating disclosure probabilities from various sharing routes. Specifically, P_{c_k} is the probability that u_t may receive u_o's photos from the propagation chain c_k. On the chain c_k, u_k is the u_t's direct sender, and hence P_{c_k} is the product of the probability $P_{u_o \Rightarrow u_k}$ that u_k receives u_o's photos and the probability $P(u_t|u_k)$ that u_k forwards the photos to u_t. Here, α is a random factor which aims to model some abnormal behavior of user u_k that u_k usually does not forward the photo to u_t suddenly decides to do so in rare cases. To achieve this, the random factor α will bring the forwarding probability $P(u_t|u_k)$ to 1 at a very low chance (i.e., 0.1%) in the probability estimation process. Next, $1 - P_{c_k}$ is the probability that u_t will not obtain u_o's photos from the chain c_k. Then, $\prod_{k=1}^{m}(1 - P_{c_k})$ is the probability that u_t will not receive u_o's photos from any of the m propagation chains. Finally, by negating the previous probability, we obtain the probability that u_t may have access to u_o's photos.

5.2 Disclosure Probability Calculation

In the previous section, we have discussed how to calculate the image disclosure probability given the possibly multiple sharing routes. The next step is to identify these sharing routes in the social network. However, the real social network is very complex which may contain a huge number of paths between two users. The critical question here is: "Is it possible to compute such image disclosure probability in practise?" The answer is positive. Even though the paths connecting two users in the social network may be huge, the number of active sharing chains is not. This is based on an important observation that people's interests in sharing others' photos typically decrease as the relationship with the photo owner becomes farther away. For example, Alice shares her photo of her first surfing with her roommate Kathy. Kathy further shares the photo with her friend Mary in the same college who may also know Alice with the thought that Mary may be surprised to see Alice is doing extreme sports. It is likely that Mary may share the photo again with other friends who may also know Alice. However, the sharing is likely to stop when it reaches a person who barely knows Alice.

Based on the above observations, we can extract a sub-network that is closely related to the photo owner before the probability calculation. The sub-network is formally defined as *personal image sharing graph* in Definition 5.

Definition 5. **(Personal Image Sharing Graph)** Given an image sharing graph SG(Ξ, SR, Ψ), the personal image sharing graph of a user u_o is PSG(Ξ_o, SR_o, Ψ_o) which satisfies the following two conditions:
(1) $\Xi_o \subseteq \Xi$, $R_o \subseteq R$, and $\Psi_o \subseteq \Psi$;
(2) $\forall\ u_j \in \Xi_o$, $\exists \psi_{u_o:u_i \rightarrow u_j}$.

The first condition in the personal image sharing graph's definition ensures that PSG is a sub-graph of the entire image sharing graph. The second condition ensures that only the users who received photos from u_o are included in this PSG. For example, reconsider the social network shown in Figure 3. We can extract the personal image sharing graph for u_o as shown in Figure 6.

Assume that the image owner u_o shares a new photo with only u_4. The red dotted arrows in Figure 6 indicate that u_1 and u_3 are u_o's contacts but are not in the sharing list of this photo. We now proceed to calculate the probability that

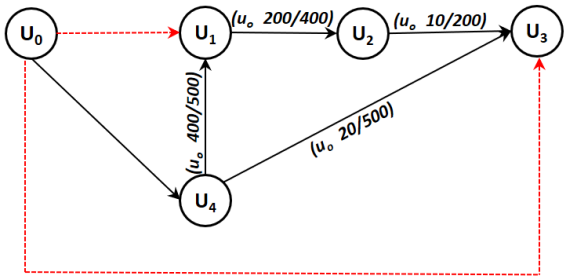


Fig. 6. **User u_o's Personal Image Sharing Graph**

two other u_o's contacts, i.e., u_1 and u_3, may also view the image.

$P_{u_o \Rightarrow u_4} = 1$

$P_{u_o \Rightarrow u_1} = P_{u_o \Rightarrow u_4} \cdot P(u_1|u_4)) = 1 \times \frac{400}{500} = 0.8$

$P_{u_o \Rightarrow u_2} = P_{u_o \Rightarrow u_1} \times \frac{200}{400} = 0.8 \times 0.5 = 0.4$

$P_{u_o \Rightarrow u_3} = 1 - (1 - P_{u_o \Rightarrow u_2} \cdot P(u_3|u_2)) \cdot (1 - P_{u_o \Rightarrow u_4} \cdot P(u_3|u_4))$

$= 1 - (1 - 0.4 \times \frac{10}{200})(1 - 1 \times \frac{20}{500}) = 0.048$

From the above example, we can see that even though u_o did not directly share the photo with u_1, there is still 80% chance that u_1 may view the photo shared from other channels. On the other hand, there is very little chance (5%) that u_3 may see the photo. To calculate these probabilities, the sequence of the node visit in the personal image sharing graph is important. The calculation sequence is u_1, u_2 and u_3 in the example. If we follow another computation order such as u_3, u_1 and u_2, we will obtain only part of the probability values for u_3, before u_2 is calculated. Once u_2's probability is known, we will have to adjust u_1's probability value. This is obviously inefficient especially in large-scale social networks. Therefore, we need to ensure that the parent nodes' probabilities are computed first. However, identifying the calculation order is not trivial due to the complicated interconnections among nodes in the social network that may create sharing loops. To efficiently and correctly calculate and aggregate the disclosure probabilities, we formally model the problem as the probability serialization (Definition 6).

Definition 6. **(Probability Serialization)** Let PSG(Ξ_o, SR_o, Ψ_o) be the personal image sharing graph of a user u_o. The probability serialization process aims to identify a serialization ordering of node visits which minimizes the node visits and ensure that each node's disclosure probability is calculated correctly. The probability serialization ordering is in the form of $u_i \succ u_{i+1} \succ \ldots \succ u_{i+k}$, where $u_i \in \Xi_o$, $\langle u_i, u_{i+1} \rangle \in SR_o$, and $u_i \succ u_{i+1}$ denotes u_i's probability will be computed before u_{i+1}'s probability.

To conduct the probability serialization, we first analyze various sharing scenarios and classify them into two main categories as shown in Figures 7 and 8, respectively. For clarity, the figures do not include the detailed sharing amounts while the arrows in the figures only indicate that there are some photos belonging to u_o being forwarded to others.

Case 1 depicts the scenario when the disclosure probability of a user needs to be calculated after all its parent nodes have been computed. Specifically, as shown in Figure 7, the photo owner u_o shares photos with his friend u_1 but not u_4. User u_1 then forwards some of the photos to u_2. User u_2 further shares the photos with u_3. Moreover, the three users u_1, u_2 and u_3 all forward some of the u_o's photos to user u_4. In this case, the probability that u_o's photos may be seen by u_4 depends on the disclosure probabilities of u_1, u_2 and u_3 which need to be computed first. The appropriate calculation order of this case is $u_1 \succ u_2 \succ u_3 \succ u_4$.

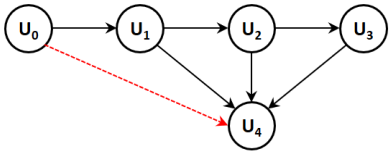


Fig. 7. **Sharing Scenario Case 1**

Case 2 depicts the scenario when there is a sharing loop. Specifically, user u_1 forwards u_o's photos to u_2, u_2 forwards the photos to u_3, and then u_3 to u_4. Without knowing that u_1 has already seen u_o's photos, u_4 forwards the photos received from u_3 to u_1, thus creating a sharing loop. In this case, even though u_4 is also u_1's immediate parent, u_1's disclosure probability does not depend on u_4 since u_4 is sharing what u_1 originally sent out. The appropriate serialization ordering of this case is $u_1 \succ u_2 \succ u_3 \succ u_4$.

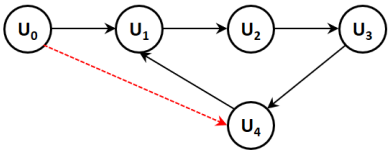


Fig. 8. **Sharing Scenario Case 2**

Based on the above classification, we now proceed to present a generic probability calculation algorithm. We employ two main data structures to facilitate the probability serialization. The first structure is a priority queue which stores the uncomputed nodes that have been visited so far. The second structure is a link list that stores the set of uncomputed parent nodes of each uncomputed node. The probability calculation takes the following steps (an outline of the algorithm is shown in Algorithm 1):

1) **Initialization**: Starting from the photo owner node u_o's initial sharing list, we look for the children nodes of the users in the sharing list and add them into the priority queue.
2) **Checking current node in the priority queue**: Then, we examine the node in the priority queue one by one. Let u_i denote the node in the priority queue that is under consideration. For any node in the priority queue, its probability is finalized only after all its parent nodes' probabilities are computed. Therefore, we check if all of u_i's parents' probabilities have already been computed. If so, we compute the probability of u_i, remove it from the priority queue

Algorithm 1 Probability Calculation Algorithm

Input: Image sharing graph
Output: Disclosure probabilities of u_o's friends
Extract u_o's personal image sharing (PIS) graph
for each user u_i in u_o's sharing list **do**
    Initialize Prob[u_i]=1
    Add u_i to priority_queue
end for
while priority_queue is not empty and U_o^- is not computed **do**
    u_i = priority_queue.pop()
    for each parent u_j of u_i **do**
        P_{ij}= chain probability (Equation 1)
        Prob[u_i]=Prob[u_i]*(1-P_{ij})
    end for
    if all of u_i's parents are computed **then**
        Prob[u_i]=1-Prob[u_i]
        Remove u_i from priority_queue
    end if
    for each u_i's direct friend u_c **do**
        if u_c is not in priority_queue **then**
            Add u_c to priority_queue
        else
            Break_Loop between u_i and u_c
        end if
    end for
end while

and perform the probability propagation routine. In the case that at least one parent node of u_i whose probability is not yet computed, we will just keep u_i in the priority queue. In both cases, we will proceed to perform the expansion routine for u_i.

3) **Probability propagation**: Given a node u_i whose probability is just computed, we will set the parent flags of all the nodes that take it as the parent to "computed" and calculate a partial probability for these nodes by plugging u_i's probability to Equation 1.
4) **Expansion**: This step is to expand the sharing chain by considering u_i's children nodes. If u_i has a child node u_c which has not been visited yet, u_c will be added to the priority queue and u_c's parents including u_i will be added to the u_c's parent list. If some of the u_c's parents' probabilities are known, their parent flags are set to "computed". After the expansion, the algorithm goes back to the second step to check the next node in the priority queue. In the case that u_c has already been stored in the priority queue, that means a sharing loop between u_i and u_c is detected. We will then give u_i a special flag which means the loop-breaking routine is pending until there is no more new node to be added to the priority queue.
5) **Breaking the Loop between u_i and u_c**: Up to this point, all of the u_i's parents should already be in the priority queue. We will compute u_c's probability by using any partial probability that u_i has so far. Note that the partial probability that u_i possesses is definitely from sources other than u_c, so it is important to factor them into u_c's probability calculation. Once u_c's probability is computed, we will remove it from the priority queue, perform the probability propagation and then check the next node in the priority queue (i.e., go back to the second step).

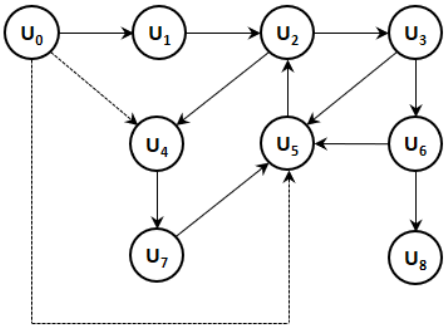


Fig. 9. **An Example of Sharing Graph**

To have a better understanding of the above probability calculation algorithm, let us step through the following example as shown in Figure 9. This example shows the image propagation from user u_o. In particular, u_o shares a new photo with u_1 but not two other friends u_4 and u_5. This example combines the two types of sharing scenarios including multi-parent relationship and multiple sharing loops.

Figure 10 presents how the information is updated in the priority queue and the parent lists throughout the probability calculation. The first black rows in the tables represent the priority queue at different steps, while the second rows represent the parent lists.

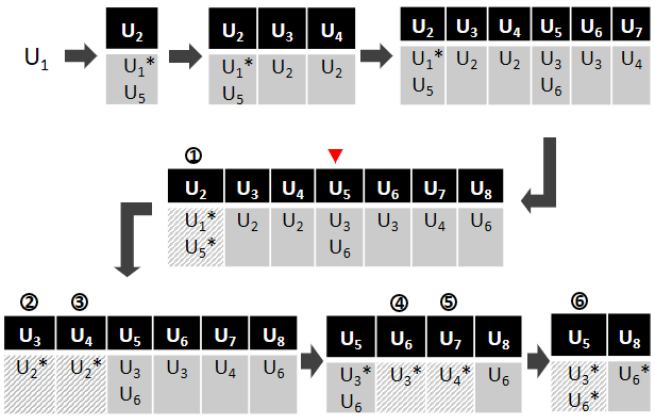


Fig. 10. **An Example of Probability Serialization**

At the beginning, the child node (u_2) of the user (u_1) who is in the photo owner's sharing list is added to the priority queue. Since u_o shares the photo directly with u_1, the probability that u_1 views the photo is 1. We start evaluating the first node in the priority queue, i.e., u_2. Since u_2 has another parent u_5 whose probability is unknown at this moment, we hold on the calculation of u_2's probability and continue expanding the sharing networks from u_2. As a result, u_2's children nodes u_3 and u_4 are added to the

priority queue too. Since u_3 and u_4 also need to wait for their parent nodes to be computed, the expansion continues whereby u_3's children (i.e., u_5 and u_6) and u_4's children (i.e., u_7) are added to the priority queue. Next, we encounter the node u_5 whose child u_2 already exists in the priority queue. That means we detect a sharing loop that involves u_2 and u_5. In this case, we give u_5 a special mark indicating that we will revisit u_5 at a later time. We continue the network expansion from u_6 to its child u_8.

Up to this point, all nodes whose probabilities can be computed should have been removed from the priority queue. It is time to deal with the sharing loops. Specifically, we locate the node u_5 which has a special mark due to the sharing loop. Then, we find the node u_2 in the priority queue which has u_5 as a parent. Since the loop starts from u_2 and goes to u_5, it is not necessary to include u_5's probability during u_2's calculation. Therefore, we go ahead to calculate u_2's probability without considering u_5. Once u_2's probability is obtained, it "unlocks" its children nodes u_3 and u_4 whose probabilities are ready for calculation too. Next, we can calculate the probabilities of u_3 and u_4's children nodes which are u_6 and u_7. Finally, we can compute u_5. Note that the calculation stops here without calculating u_8 because all of u_o's contacts in the non-sharing list have been computed. The complete probability calculation ordering is $u_2 \succ u_3 \succ u_4 \succ u_6 \succ u_7 \succ u_5$ (indicated by the circled number on top of each node in the figure).

The above probability calculation algorithm provides the calculation ordering for all the users that are in the photo owner u_o's personal image sharing graph. It is worth noting that the efficiency of probability calculation can be further improved by stopping the calculation for a node if its current probability is already higher than the decision threshold. For example, if through currently explored sharing chains, the disclosure probability is as high as 99%, it is not necessary to keep checking remaining sharing routes.

The complexity of our probability calculation algorithm is O(n) as each node in the personal image sharing graph is first visited once during the personal image sharing graph extraction and then calculated once in the priority queue. It is worth noting that the probability calculation works the same for different categories of images. When multiple categories of image information is available through the image classification tool, we still just need to construct one image sharing graph for each user. The only difference will be the information stored at each node in the sharing graph. Specifically, on each node, there will be multiple tuples, each of which corresponds to a category of image sharing statistics. Since an image only belongs to one category, the calculation of a single image will only access its corresponding statistic information at the nodes, and hence there will not be any impact on the calculation efficiency.

5.3 Privacy Harmonization among Multiple Users

In the previous sections, we have discussed how to handle a photo with a single owner. Indeed, the risk estimation algorithm can be easily extended to address the policy harmonization issues occurring in a photo with multiple owners. It is common that different users may have different privacy preferences regarding the same photo. Consider the example when there is a group photo of Alice, Bob and Mary. Alice would like to share the photo with her family members only, while both Bob and Mary would like to share the photo with their close friends. It is possible that some of Bob and Mary's close friends are also Alice's friends who will be able to view Alice's photo although Alice's initial intention is to share only within her family. Our goal is to estimate the risk of privacy breach due to such difference. Our system will calculate the disclosure probability of the photo being seen by people who are wanted by Alice due to the sharing activities from Bob and Mary. We will present the estimated risk to all the photo owners so that they can refine their privacy policies.

In order to achieve the above goal, instead of calculating disclosure probabilities for an individual photo owner as discussed in the previous sections, we need to calculate the following disclosure matrix.

Definition 7. **(Disclosure Matrix)** Let u_1, ..., u_n denote the group of people depicted in a photo img, and Pol_1, ..., Pol_n denote the policies belonging to each photo owner, respectively. The disclosure matrix is defined below, where $u_{i_j} \in \bigcup_{w=1}^{n} U_w^+ / \{u_1, ...u_n\}$.

$$\begin{array}{c} \\ u_1 \\ u_2 \\ \cdots \\ u_n \end{array} \begin{array}{c} \begin{array}{cccc} u_{i_1} & u_{i_2} & \cdots & u_{i_k} \end{array} \\ \begin{bmatrix} P(U_1^-|u_{i_1}) & P(U_1^-|u_{i_2}) & \cdots & P(U_1^-|u_{i_k}) \\ P(U_2^-|u_{i_1}) & P(U_2^-|u_{i_2}) & \cdots & P(U_2^-|u_{i_k}) \\ \cdots & \cdots & \cdots & \cdots \\ P(U_n^-|u_{i_1}) & \cdots & \cdots & P(U_n^-|u_{i_k}) \end{bmatrix} \end{array}$$

The main idea underlying the disclosure matrix is to check the potential privacy breach that may be caused by the union of the groups of people in all the photo owners' sharing list. After the calculating the disclosure matrix, we will identify and suggest the photo owners to remove potentially high-risk sharing activities. The following is an illustrating example.

Suppose that a photo has three owners: u_1, u_2 and u_3. The sharing lists in the photo owners' policies are the following:

$$\begin{aligned} \text{Pol}_{u1} &= \{u_1, u_2, u_3, u_4, u_5\} \\ \text{Pol}_{u2} &= \{u_1, u_2, u_3, u_4, u_6\} \\ \text{Pol}_{u3} &= \{u_1, u_2, u_3, u_5, u_7\} \end{aligned}$$

The corresponding disclosure matrix considers the unions of the sharing list excluding the photo owners themselves who are assumed to have full access to the photo. Assume that we obtain the probabilities as shown in the following:

$$\begin{array}{c} \\ u_1 \\ u_2 \\ u_3 \end{array} \begin{array}{c} \begin{array}{cccc} u_4 & u_5 & u_6 & u_7 \end{array} \\ \begin{bmatrix} 0.1 & \mathbf{0.9} & 0.05 & 0 \\ 0.1 & \mathbf{0.95} & \mathbf{0.8} & 0.1 \\ 0 & \mathbf{0.85} & 0.2 & 0.3 \end{bmatrix} \end{array}$$

From the above disclosure matrix, we can see that $P(U_1^-|u_5)$, $P(U_2^-|u_5)$, and $P(U_3^-|u_5)$ are very high (i.e., above a given privacy threshold), which means the risk that people in the non-sharing lists of all the photo owners

may see this photo due to the further propagation from u_5. Therefore, our REMIND system will suggest all the photo owners to remove u_5 from their sharing list. In addition, user u_2's sharing with u_6 may cause potential privacy breach for him/herself, thus, we would suggest u_2 to remove u_6 from the sharing list. If all the users agree with suggestions, the policy harmonization will result in the following new policies:

$$\text{Pol'}_{u1} = \{u_1, u_2, u_3, u_4\}$$
$$\text{Pol'}_{u2} = \{u_1, u_2, u_3, u_4\}$$
$$\text{Pol'}_{u3} = \{u_1, u_2, u_3, u_7\}$$

6 Experimental Study

In this section, we present our experimental studies that evaluate both effectiveness and efficiency of our proposed approach. Specifically, we conducted user studies to see how people would react when presented a probability score of their privacy breach as computed by our system. The goal is to validate the usefulness of our proposed REMIND system. Next, we tested the performance of our system by using real social network datasets with various sharing scenarios. The second set of experiments aims to validate the efficiency of our proposed system.

6.1 Effectiveness Study

While we have implemented a prototype of the proposed REMIND system, we could not evaluate it in the real social network settings since its deployment in the real world requires the installation at the service provider side, e.g., installed as an additional function by the Facebook, so as to gain the access to the image sharing history. Thus, we built a simulated social network environment and conducted a A/B test as follows.

- Environment A is the one without the REMIND system which is similar to the existing social networks where people share images as usual. Specifically, a user is presented with an image and corresponding background story of the image so that the user can feel more personal about the image. There are total 10 scenarios and each scenario contains two images targeting female and male participants, respectively. The 10 scenarios aim to cover common cases where people may have privacy concerns, such as funny costumes, crazy parties, family vacations, selfie of bad mood, surprising gift, casual time at home, sickness, dangerous sports, and risky adventure. Then, the user is asked to select one or more groups of users that they would like to share the image. We provide six common groups for the users to choose, which are close family members, relatives, close friends, friends, co-workers, and boss. This mimic the common practice in the real social networks.
- Environment B is the one with the REMIND function. The difference from Environment A is that after the same user chose the group of people to share, we present the probability of privacy breach (if higher than a threshold say 90%) to the user and ask if they would like to change their initial privacy settings.

We recruited another 183 participants on campus and online. There are 99 males and 84 females. The age distribution is: 20% between 18 and 20, 49% between 21 and 30, 21% between 31 and 40, and 20% older than 41. All of the participants have at least one social media account and have experience of sharing photos.

Through the simulation, we have the following interesting findings. The adoption of the risk reminder depends on both the type of the image to be shared and the initial sharing list that the user has chosen. In general, the more sensitive the image is, the higher the chance the user will accept the REMIND system's recommendation of changing their original privacy settings. For example, the images that depict funny, crazy or sadness moments are typically considered sensitive and usually shared with close family members and friends. For such kind of images, when there is an alert about potential disclosure to the user's boss, around 75% of the participants chose to accept the REMIND system's suggestion of removing the person who may cause this breach from the share list. As for image of seeking gift ideas, about 80% of the participants do not want to take any risk of the gift being leaked to the recipients and agreed with the REMIND system to remove potential causes. When it comes to images of someone trying out new things (e.g., smoking hookah) that may have debatable opinions in the public, the photo owners seem to be more concerned about the potential privacy breach. As a result, more than 90% of the participants chose to accept the REMIND system's suggestion to achieve better privacy protection in that case.

When the images are less sensitive such as casual photos taken at home, the decisions of whether accepting the privacy alert split. Some users still want to limit the photos to be seen by a desired group of people, but some do not. Specifically, when the user initially shares the image with family members, about 46% of them do not care if the image may also be viewed by friends; when the user initially shared the image with friends, about 65% of them do not care if the image may also be viewed by their relatives; when the user initially shared the images with co-workers, about 60% of them do not worry about that photo being seen by their bosses.

To sum up, among total 1444 privacy alerts issued in our simulated social networking environment, 60% are accepted by the participants, which means 60% of existing privacy configurations may be further improved. This shows the potential of adoption of our REMIND system in the real world.

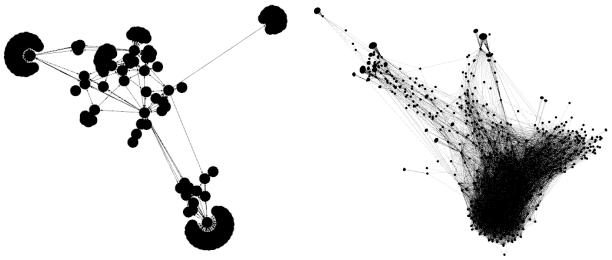
6.2 Efficiency Study

We now proceed to evaluate the efficiency of our approach. Since our probability model looks into large-scale historical image sharing data and convoluted social networks, it is critical that the disclosure probability can be computed in a real-time manner to provide the users an immediate reminder when they are uploading new photos.

To examine the efficiency, we test our approach in real social networks released by Facebook and Twitter [32]. Table 3 presents the statistics of the two social networks, and Figures **??** and 11 depict the network graphs where the black dots represent users and lines represent the connections between users. We can observe that these real social

TABLE 3
Real Social Network Datasets

Dataset	Facebook	Twitter
Total number of nodes	3,908	81,306
Total number of edges	168,194	1,768,149
Average degree	43	21
Maximum degree	293	1635



(a) Facebook (b) Twitter

Fig. 11. **Facebook and Twitter Networks**

networks are very complicated and nothing close to uniform distribution. We study the effects of their structures and sizes on the computation efficiency.

Since current social media sites only release the social network connections, but not the image sharing statistics yet, we simulate a variety of scenarios in terms of image sharing on these real social networks as described in Definition 4. It is worth noting that although the image sharing statistic information is synthetic, it does not affect the efficiency test since the social network topology is real and our sharing parameters cover a wide range of possible sharing scenarios. Specifically, we first generate a random number of photos ranging from 100 to 1000 for each user. Then, for each user, we randomly select a subset of his/her friends to share certain percentage of the photos, and the size of this subset is varied in the following experiments. The receivers of the shared photo will forward a random number of received photos to a random number of their friends. In this way, the photos are propagated in the social network similar to the real world scenario. We control the propagation by setting the maximum number of hops to forward the photos since a personal photo may not be interesting to people who have almost no relationship with the photo owner. We vary the number of people in the initial sharing list. We also vary the speed of image sharing convergence as a photo may becomes less interesting to people who are farther away from the photo owner. Besides real social networks, we also test the synthetic networks with more than 20 million nodes to evaluate the scalability of our algorithm. The following subsections elaborate the detailed experimental settings for each round of experiments and report the corresponding results. All the experiments were conducted in a computer with Intel Core i7-7700K CPU (4.20 GHz) and 16GB RAM.

6.2.1 *Effect of the Number of Propagation Hops*

In the first round of experiments, we evaluate the effect of the number of image propagation hops ranging from 1 to 5. When there is only one hop, the photo owners share the photos with their direct friends and their friends will not forward the photos to anyone else. When there are five hops, the photos will be forwarded by the photo owners' friends to the friends' friends until 5 hops. The reason to choose maximum 5 hops is based on the "six degrees of separation" theory [33] that any two users can be connected through 5 acquaintances, and we choose one degree less to avoid the photos being propagated in the whole social networks which loses the privacy protection sense. Moreover, in social network, the average degree of separation is only 3.5 as reported by a study [34]. Therefore, we chose 3 hops as the default values for the subsequent tests.

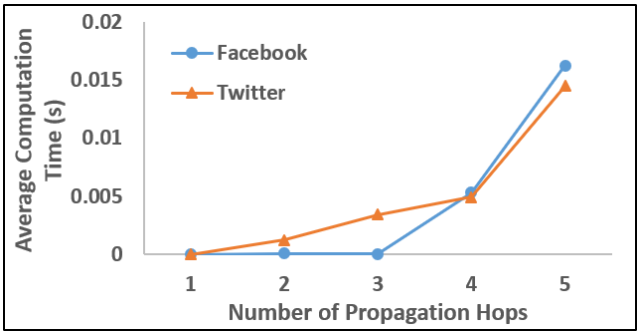


Fig. 12. **Effect of the Number of the Hops**

Figure 12 reports the average time taken to compute the disclosure probability of a photo owner's friend who is not in the initial sharing list. We can observe that the calculation takes less than 1s in all cases for both the Facebook and Twitter datasets. The efficiency could be attributed to the extraction of the personal sharing graphs as well as the probability serialization algorithm, both of which help reduce the amount of users (nodes in the social network) to be examined and calculated. Moreover, we also observe that the calculation time increases when the photos are propagated through more hops. The reason is that the more hops, the more users may receive the shared photos, resulting in various sharing chains and loops which takes time to calculate. Actually, the average disclosure probability of the friends who are not in the initial sharing list also increases with the hops.

6.2.2 *Effect of the Number of Friends in the Initial Sharing List*

In this round of experiments, we fix the image propagation hops to 3 and vary the number of friends in the initial sharing list from 50 to 200. As shown in Figure 13, the average time to calculate the disclosure probability for a user in both datasets can be done in just a few milliseconds. This again proves the efficiency of our algorithm. In addition, we also observe that the calculation time increases with the size of the sharing list. This is because the more people in the initial sharing list, the wider audience the photos may reach, which leads to a complicated sharing graph. As a result, there may be more ancestor nodes to be computed before finalizing a user's disclosure probability. Note that the wider audience also means the corresponding increase in the average disclosure probabilities.

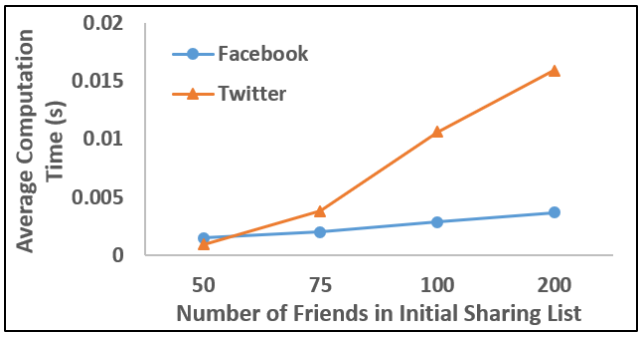


Fig. 13. **Effect of the Size of the Initial Sharing List**

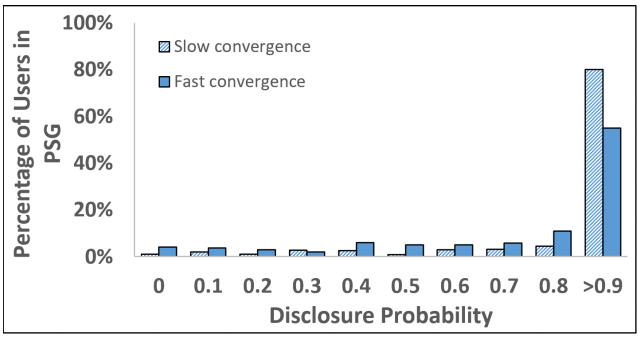


Fig. 15. **Probability Distribution**

6.2.3 *Effect of the Sharing Convergence Speed*

We also evaluate the effect of the sharing convergence speed. We simulate this by decreasing the number of friends to share the photos at each hop. Specifically, the statistic sharing information is generated by allowing each user to share the photos with 75 friends. For each friend who received the photo, he/she forwards the photo to a smaller number of friends, e.g., 20% less of the previous hop. The sharing stops when reaching the 3rd hop. Figure 14 shows the average probability calculation time for each user. Observe that the calculation time decreases when the sharing convergence speed increases. This is because the number of people in the sharing list at each hop decreases, and hence the overall size of the sharing graph decreases too. In other words, the smaller the scope of the sharing, the faster the calculation.

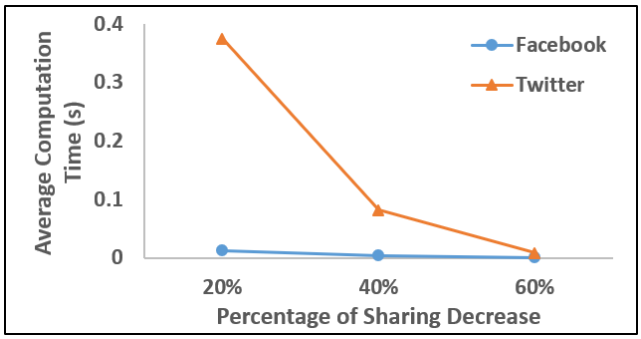


Fig. 14. **Effect of Sharing Convergence Speed**

Also, the smaller the sharing scope, the lower the average disclosure probabilities. For example, let us take a closer look at the probability distribution of the Facebook dataset. When the convergence speed is decreasing by 20% per hop, there are about 80% of people in the photo owner's personal image sharing graph (including those in the initial sharing list) may see the photo with probability higher than 0.9 (denoted as "Slow convergence" in Figure 15); when the convergence speed is faster (i.e., 60%), the number of people with high disclosure probability drops to 50% (denoted as "Fast convergence" in Figure 15)

6.2.4 *Large-Scale Testing*

Finally, we evaluate the scalability of our proposed algorithm by using synthetically generated large-scale datasets. Figure 16 shows the average probability calculation time for an image when the total number of nodes in the synthetic social network increases from 1K to 20M. The number of hops for the image propagation is set to the default value 3, and each user forwards the images to 15 randomly selected friends. From the figure, we can observe that the calculation time only increases slightly with the total number of nodes in the social network. This again indicates the advantage of our proposed personal sharing graph which does not increase due to the increase of the social network size. In other words, as long as the user's contacts and image sharing behavior stay the same, the calculation scope (i.e., extracted personal sharing graph) is similar for the user no matter the user is in a small social network or a large social network. This result also demonstrates the scalability of our approach.

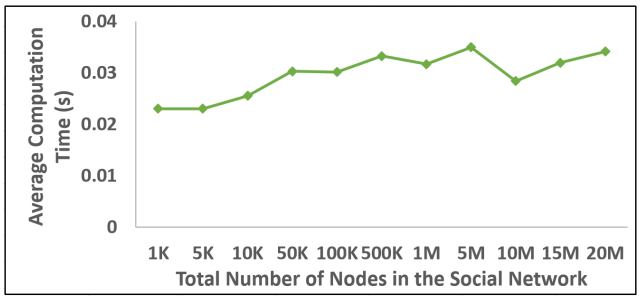


Fig. 16. **Effect of Total Number of Nodes in the Social Network**

In addition, we also examine an extreme case when there are a small number of users with an extremely large number of contacts in the social network. To simulate this scenario, we randomly select 10,000 users from a 100K-node social network to be the contacts of the photo owner who then randomly selects 1% (100 users) of his contacts to share the images. Among the selected 1% of his contacts, we again randomly select a user to have 10,000 contacts while other contacts only have a few hundred contacts as that in Facebook. We simulate this for 3 hops of propagation and then test the calculation time. The average time to calculate the disclosure probability to a person takes just 6ms. Since the number of contacts who are not in the initial sharing list of the photo owner is large, the total time to calculate the disclosure alert for an image for the photo owner takes about 58s. The calculation time may be further shortened by considering the use of parallel computing for the multiple contacts at the same time, for which we will explore as our future work.

7 Conclusion

In this paper, we present a novel risk reminder system that offers the social network users a quantitative view of their image sharing risks due to friend-to-friend re-sharing. Our proposed REMIND system is based on a sophisticated probability model that models the large-scale image sharing statistic information and captures the complicated sharing propagation chains and loops. Our system also addresses the policy harmonization challenges in multi-owner photos. We have carried out both user studies and performance studies to validate the effectiveness and efficiency of our approach.

Acknowledgements

The work of Dan Lin, Douglas Steiert and Joshua Morris is funded by the National Science Foundation under the project CNS-1651455 and DGE-1433659. The work of Jianping Fan is funded by the National Science Foundation under the project CNS-1651166. Work from Anna Squicciarini was partially funded by National Science Foundation under grant 1453080.

References

[1] C. G. Akcora, B. Carminati, and E. Ferrari, "Risks of friendships on social networks," in *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, 2012, pp. 810–815.

[2] N. Laleh, B. Carminati, and E. Ferrari, "Graph based local risk estimation in large scale online social networks," in *IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, 2015, pp. 528–535.

[3] O. Kafalı, A. Günay, and P. Yolum, "Detecting and predicting privacy violations in online social networks," *Distributed and Parallel Databases*, vol. 32, no. 1, pp. 161–190, Mar 2014.

[4] N. Kokciyan and P. Yolum, "Priguard: A semantic approach to detect privacy violations in online social networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 10, pp. 2724–2737, 2016.

[5] J. Bonneau, J. Anderson, and G. Danezis, "Prying data out of a social network," in *Social Network Analysis and Mining, 2009. ASONAM'09. International Conference on Advances in*. IEEE, 2009, pp. 249–254.

[6] L. Fang and K. LeFevre, "Privacy wizards for social networking sites," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 351–360.

[7] A. Mazzia, K. LeFevre, and E. Adar, "The pviz comprehension tool for social network privacy settings," in *Proceedings of the Eighth Symposium on Usable Privacy and Security*. ACM, 2012, p. 13.

[8] R. G. Pensa and G. Di Blasi, "A semi-supervised approach to measuring user privacy in online social networks," in *Discovery Science*, T. Calders, M. Ceci, and D. Malerba, Eds., Cham, 2016, pp. 392–407.

[9] A. C. Squicciarini, S. Sundareswaran, D. Lin, and J. Wede, "A3p: Adaptive policy prediction for shared images over popular content sharing sites," in *Proceedings of the 22Nd ACM Conference on Hypertext and Hypermedia*, 2011, pp. 261–270.

[10] J. M. Such and N. Criado, "Resolving multi-party privacy conflicts in social media," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1851–1863, 2016.

[11] T. Mariotti, "How can we track when someone shares through the facebook share (not like) button?" in *https://www.quora.com/How-can-we-track-when-someone-shares-through-the-Facebook-Share-not-Like-button*, 2012.

[12] E. Protalinski, "13 million us facebook users don't change privacy settings," in *https://www.zdnet.com/article/13-million-us-facebook-users-dont-change-privacy-settings/*.

[13] F. Adu-Oppong, C. K. Gardiner, A. Kapadia, and P. P. Tsang, "Social circles: Tackling privacy in social networks," in *Symposium on Usable Privacy and Security (SOUPS)*, 2008.

[14] A. Squicciarini, S. Karumanchi, D. Lin, and N. Desisto, "Identifying hidden social circles for advanced privacy configuration," *Computers and Security*, vol. 41, pp. 40–51, 2014.

[15] P. Klemperer, Y. Liang, M. Mazurek, M. Sleeper, B. Ur, L. Bauer, L. F. Cranor, N. Gupta, and M. Reiter, "Tag, you can see it!: Using tags for access control in photo sharing," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2012, pp. 377–386.

[16] E. Spyromitros-Xioufis, S. Papadopoulos, A. Popescu, and Y. Kompatsiaris, "Personalized privacy-aware image classification," in *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*. ACM, 2016, pp. 71–78.

[17] J. Yu, B. Zhang, Z. Kuang, D. Lin, and J. Fan, "iprivacy: Image privacy protection by identifying sensitive objects via deep multi-task learning," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 5, pp. 1005–1016, 2017.

[18] A. C. Squicciarini, D. Lin, S. Sundareswaran, and J. Wede, "Privacy policy inference of user-uploaded images on content sharing sites," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 1, pp. 193–206, 2015.

[19] J. Yu, Z. Kuang, B. Zhang, W. Zhang, D. Lin, and J. Fan, "Leveraging content sensitiveness and user trustworthiness to recommend fine-grained privacy settings for social image sharing," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 5, pp. 1317–1332, 2018.

[20] Y. Li, Y. Li, Q. Yan, and R. H. Deng, "Privacy leakage analysis in online social networks," *Computers & Security*, vol. 49, pp. 239–254, 2015.

[21] A. Acquisti, I. Adjerid, R. Balebako, L. Brandimarte, L. F. Cranor, S. Komanduri, P. G. Leon, N. Sadeh, F. Schaub, M. Sleeper, Y. Wang, and S. Wilson, "Nudges for privacy and security: Understanding and assisting users’ choices online," *ACM Computing Surveys*, vol. 50, no. 3, pp. 44:1–44:41, Aug. 2017.

[22] K. Liu and E. Terzi, "A framework for computing the privacy scores of users in online social networks," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 5, no. 1, p. 6, 2010.

[23] A. Srivastava and G. Geethakumari, "Measuring privacy leaks in online social networks," in *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, 2013, pp. 2095–2100.

[24] G. Petkos, S. Papadopoulos, and Y. Kompatsiaris, "Pscore: a framework for enhancing privacy awareness in online social networks," in *2015 10th International Conference on Availability, Reliability and Security*. IEEE, 2015, pp. 592–600.

[25] M. Kimura and K. Saito, "Tractable models for information diffusion in social networks," in *European conference on principles of data mining and knowledge discovery*. Springer, 2006, pp. 259–271.

[26] E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic, "The role of social networks in information diffusion," in *Proceedings of the 21st international conference on World Wide Web*. ACM, 2012, pp. 519–528.

[27] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, "Information diffusion in online social networks: A survey," *ACM Sigmod Record*, vol. 42, no. 2, pp. 17–28, 2013.

[28] H. Hu, G.-J. Ahn, and J. Jorgensen, "Multiparty access control for online social networks: model and mechanisms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 7, pp. 1614–1627, 2013.

[29] N. Kökciyan, N. Yaglikci, and P. Yolum, "An argumentation approach for resolving privacy disputes in online social networks," *ACM Transactions on Internet Technology (TOIT)*, vol. 17, no. 3, p. 27, 2017.

[30] D. Kekulluoglu, N. Kokciyan, and P. Yolum, "Preserving privacy as social responsibility in online social networks," *ACM Transactions on Internet Technology (TOIT)*, vol. 18, no. 4, p. 42, 2018.

[31] P. Rao, D. Lin, E. Bertino, N. Li, and J. Lobo, "Fine-grained integration of access control policies," *Computers and Security*, vol. 30, no. 2, pp. 91 – 107, 2011, special Issue on Access Control Methods and Technologies. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404810000891

[32] J. Leskovec, "Stanford large network dataset collection," in *https://snap.stanford.edu/data/*.

[33] Wikipedia, "Six degree of separation," in *https://en.wikipedia.org/wiki/Six_degrees_of_separation*.

[34] S. Bhagat, M. Burke, C. Diuk, I. O. Filiz, and S. Edunov, "Three and a half degrees of separation," in *https://research.fb.com/three-and-a-half-degrees-of-separation/*.