Privacy Policy Inference of User-Uploaded Images on Content Sharing Sites

Anna Cinzia Squicciarini, *Member, IEEE*, Dan Lin, Smitha Sundareswaran, and Joshua Wede

Abstract—With the increasing volume of images users share through social sites, maintaining privacy has become a major problem, as demonstrated by a recent wave of publicized incidents where users inadvertently shared personal information. In light of these incidents, the need of tools to help users control access to their shared content is apparent. Toward addressing this need, we propose an Adaptive Privacy Policy Prediction (A3P) system to help users compose privacy settings for their images. We examine the role of social context, image content, and metadata as possible indicators of users' privacy preferences. We propose a two-level framework which according to the user's available history on the site, determines the best available privacy policy for the user's images being uploaded. Our solution relies on an image classification framework for image categories which may be associated with similar policies, and on a policy prediction algorithm to automatically generate a policy for each newly uploaded image, also according to users' social features. Over time, the generated policies will follow the evolution of users' privacy attitude. We provide the results of our extensive evaluation over 5,000 policies, which demonstrate the effectiveness of our system, with prediction accuracies over 90 percent.

Index Terms—Online information services, web-based services

1 Introduction

Images are now one of the key enablers of users' connectivity. Sharing takes place both among previously established groups of known people or social circles (e. g., Google+, Flickr or Picasa), and also increasingly with people outside the users social circles, for purposes of social discovery-to help them identify new peers and learn about peers interests and social surroundings. However, semantically rich images may reveal content-sensitive information []. Consider a photo of a students 2012 graduationceremony, for example. It could be shared within a Google+ circle or Flickr group, but may unnecessarily expose the studentsBApos familymembers and other friends. Sharing images within online content sharing sites,therefore,may quickly leadto unwanted disclosure and privacy violations [3], [24]. Further, the persistent nature of online media makes it possible for other users to collect rich aggregated information about the owner of the published content and the subjects in the published content [3], [20], [24]. The aggregated information can result in unexpected exposure of one's social environment and lead to abuse of one's personal information.

- *A.C. Squicciarini is with the College of Information Sciences & Technology, The Pennsylvania State University, University Park, PA. E-mail: asquicciarini@ist.psu.edu.*
- *D. Lin is with the Department of Computer Science, Missouri University of Science & Technology, Rolla, MO. E-mail: lindan@mst.edu.*
- *S. Sundareswaran is with the College of Information Sciences & Technology, The Pennsylvania State University, University Park, PA. E-mail: sus263@psu.edu.*
- *J. Wede is with the Department of Psychology, The Pennsylvania State University, University Park, PA. E-mail: jwede@psu.edu.*

Manuscript received 29 Jul. 2013; revised 14 Jan. 2014; accepted 8 Apr. 2014. Date of publication 28 Apr. 2014; date of current version 1 Dec. 2014. Recommended for acceptance by E. Ferrari.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TKDE.2014.2320729

Most content sharing websites allow users to enter their privacy preferences. Unfortunately, recent studies have shown that users struggle to set up and maintain such privacy settings [1], [11], [22], [33]. One of the main reasons provided is that given the amount of shared information this process can be tedious and error-prone. Therefore, many have acknowledged the need of policy recommendation systems which can assist users to easily and properly configure privacy settings [7], [22], [28], [30]. However, existing proposals for automating privacy settings appear to be inadequate to address the unique privacy needs of images [3], [5], [41], due to the amount of information implicitly carried within images, and their relationship with the online environment wherein they are exposed.

In this paper, we propose an *Adaptive Privacy Policy Prediction* (A3P) system which aims to provide users a hassle free privacy settings experience by automatically generating personalized policies. The A3P system handles user uploaded images, and factors in the following criteria that influence one's privacy settings of images:

- *The impact of social environment and personal characteristics*. Social context of users, such as their profile information and relationships with others may provide useful information regarding users' privacy preferences. For example, users interested in photography may like to share their photos with other amateur photographers. Users who have several family members among their social contacts may share with them pictures related to family events. However, using common policies across all users or across users with similar traits may be too simplistic and not satisfy individual preferences. Users may have drastically different opinions even on the same type of images. For example, a privacy adverse person may be willing to share all his personal images while a more conservative person may just want to share personal images

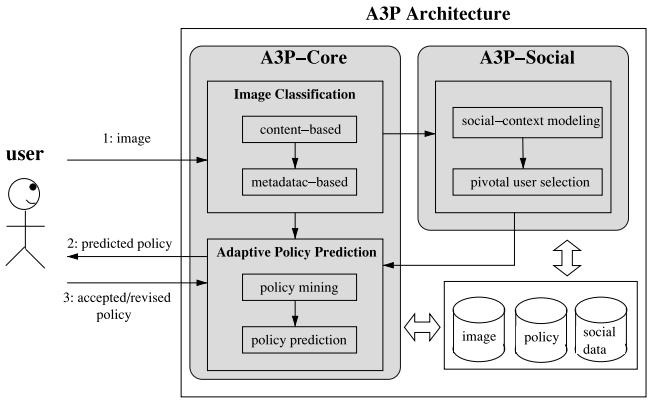

Fig. 1. System overview.

with his family members. In light of these considerations, it is important to find the balancing point between the impact of social environment and users' individual characteristics in order to predict the policies that match each individual's needs.

Moreover, individuals may change their overall attitude toward privacy as time passes. In order to develop a personalized policy recommendation system, such changes on privacy opinions should be carefully considered.

- *The role of image's content and metadata*. In general, similar images often incur similar privacy preferences, especially when people appear in the images. For example, one may upload several photos of his kids and specify that only his family members are allowed to see these photos. He may upload some other photos of landscapes which he took as a hobby and for these photos, he may set privacy preference allowing anyone to view and comment the photos.

  Analyzing the visual content may not be sufficient to capture users' privacy preferences. Tags and other metadata are indicative of the social context of the image, including where it was taken and why [4], and also provide a synthetic description of images, complementing the information obtained from visual content analysis.

Corresponding to the aforementioned two criteria, the proposed A3P system is comprised of two main building blocks (as shown in Fig. 1): A3P-Social and A3P-Core. The A3P-core focuses on analyzing each individual user's own images and metadata, while the A3P-Social offers a community perspective of privacy setting recommendations for a user's potential privacy improvement. We design the interaction flows between the two building blocks to balance the benefits from meeting personal characteristics and obtaining community advice.

To assess the practical value of our approach, we built a system prototype and performed an extensive experimental evaluation. We collected and tested over 5,500 real policies generated by more than 160 users. Our experimental results demonstrate both efficiency and high prediction accuracy of our system.

A preliminary discussion of the A3P-core was presented in [32]. In this work, we present an overhauled version of A3P, which includes an extended policy prediction algorithm in A3P-core (that is now parameterized based on user groups and also factors in possible outliers), and a new A3P-social module that develops the notion of social context to refine and extend the prediction power of our system. We also conduct additional experiments with a new data set collecting over 1,400 images and corresponding policies, and we extend our analysis of the empirical results to unveil more insights of our system's performance.

The rest of the paper is organized as follows. Section 2 reviews related works. Section 3 introduces preliminary notions. Section 4 introduces the A3P-core and Section 5 introduces the A3P-Social. Section 6 reports the experimental evaluation. Finally, Section 7 concludes the paper.

2 Related Work

Our work is related to works on privacy setting configuration in social sites, recommendation systems, and privacy analysis of online images.

2.1 Privacy Setting Configuration

Several recent works have studied how to automate the task of privacy settings (e.g., [7], [15], [20], [22], [27], [28]).

Bonneau et al. [7] proposed the concept of privacy suites which recommend to users a suite of privacy settings that "expert" users or other trusted friends have already set, so that normal users can either directly choose a setting or only need to do minor modification. Similarly, Danezis [8] proposed a machine-learning based approach to automatically extract privacy settings from the social context within which the data is produced. Parallel to the work of Danezis, Adu-Oppong et al. [15] develop privacy settings based on a concept of "Social Circles" which consist of clusters of friends formed by partitioning users' friend lists. Ravichandran et al. [30] studied how to predict a user's privacy preferences for location-based data (i.e., share her location or not) based on location and time of day. Fang et al. [28] proposed a privacy wizard to help users grant privileges to their friends. The wizard asks users to first assign privacy labels to selected friends, and then uses this as input to construct a classifier which classifies friends based on their profiles and automatically assign privacy labels to the unlabeled friends. More recently, Klemperer et al. [20] studied whether the keywords and captions with which users tag their photos can be used to help users more intuitively create and maintain access-control policies. Their findings are inline with our approach: tags created for organizational purposes can be repurposed to help create reasonably accurate access-control rules.

The aforementioned approaches focus on deriving policy settings for only traits, so they mainly consider social context such as one's friend list. While interesting, they may not be sufficient to address challenges brought by image files for which privacy may vary substantially not just because of social context but also due to the actual image content. As far as images, authors in [41] have presented an expressive language for images uploaded in social sites. This work is complementary to ours as we do not deal with policy expressiveness, but rely on common forms policy specification for our predictive algorithm.

In addition, there is a large body of work on image content analysis, for classification and interpretation (e.g., [14], [37], [46]), retrieval ([12], [13] are some examples), and photo ranking [35], [40], also in the context of online photo sharing sites, such as Flickr [10], [29], [36]. Of these works, Zerr's work [43] is probably the closest to ours. Zerr explores privacy-aware image classification using a mixed set of features, both content and meta-data. This is however a binary classification (private versus public), so the classification task is very different than ours. Also, the authors do not deal with the issue of cold-start problem.

2.2 Recommendation Systems

Our work is related to some existing recommendation systems which employ machine learning techniques.

Chen et al. [9] proposed a system named SheepDog to automatically insert photos into appropriate groups and recommend suitable tags for users on Flickr. They adopt concept detection to predict relevant concepts (tags) of a photo. Choudhury et al. [10] proposed a recommendation framework to connect image content with communities in online social media. They characterize images through three types of features: visual features, user generated text tags, and social interaction, from which they recommend the most likely groups for a given image. Similarly, Yu et al. [42] proposed an automated recommendation system for a user's images to suggest suitable photo-sharing groups.

There is also a large body of work on the customization and personalization of tag-based information retrieval (e.g., [21], [23], [45]), which utilizes techniques such as association rule mining. For example, [45] proposes an interesting experimental evaluation of several collaborative filtering algorithms to recommend groups for Flickr users. These approaches have a totally different goal to our approach as they focus on sharing rather than protecting the content.

3 A3P Framework

3.1 Preliminary Notions

Users can express their privacy preferences about their content disclosure preferences with their socially connected users via privacy policies. We define privacy policies according to Definition 1. Our policies are inspired by popular content sharing sites (i.e., Facebook, Picasa, Flickr), although the actual implementation depends on the specific content-management site structure and implementation.

Definition 1. *A privacy policy P of user u consists of the following components:*

- *Subject (S): A set of users socially connected to u.*
- *Data (D): A set of data items shared by u.*
- *Action (A): A set of actions granted by u to S on D.*
- *Condition (C): A boolean expression which must be satisfied in order to perform the granted actions.*

In the definition, users in S can be represented by their identities, roles (e.g., family, friend, coworkers), or organizations (e.g., non-profit organization, profit organization). D will be the set of images in the user's profile. Each image has a unique ID along with some associated metadata like tags "vacation", "birthday". Images can be further grouped into albums. As for A, we consider four common types of actions: *{view, comment, tag, download}*. Last, the condition component C specifies when the granted action is effective. C is a Boolean expression on the grantees' attributes like time, location, and age. For better understanding, an example policy is given below.

Example 1. Alice would like to allow her friends and coworkers to comment and tag images in the album named "vacation_album" and the image named "summer.jpg" before year 2012. Her privacy preferences can be expressed by the following policy:

P: [{friend, coworker}, {vacation_album, summer.jpg}, {comment, tag}, (date $<$ 2012)].

3.2 System Overview

The A3P system consists of two main components: A3P-core and A3P-social. The overall data flow is the following. When a user uploads an image, the image will be first sent to the A3P-core. The A3P-core classifies the image and determines whether there is a need to invoke the A3P-social. In most cases, the A3P-core predicts policies for the users directly based on their historical behavior. If one of the following two cases is verified true, A3P-core will invoke A3P-social: (i) The user does not have enough data for the type of the uploaded image to conduct policy prediction; (ii) The A3P-core detects the recent major changes among the user's community about their privacy practices along with user's increase of social networking activities (addition of new friends, new posts on one's profile etc). In above cases, it would be beneficial to report to the user the latest privacy practice of social communities that have similar background as the user. The A3P-social groups users into social communities with similar social context and privacy preferences, and continuously monitors the social groups. When the A3P-social is invoked, it automatically identifies the social group for the user and sends back the information about the group to the A3P-core for policy prediction. At the end, the predicted policy will be displayed to the user. If the user is fully satisfied by the predicted policy, he or she can just accept it. Otherwise, the user can choose to revise the policy. The actual policy will be stored in the policy repository of the system for the policy prediction of future uploads.

4 A3P-Core

There are two major components in A3P-core: (i) Image classification and (ii) Adaptive policy prediction. For each user, his/her images are first classified based on content and metadata. Then, privacy policies of each category of images are analyzed for the policy prediction.

Adopting a two-stage approach is more suitable for policy recommendation than applying the common one-stage data mining approaches to mine both image features and policies together. Recall that when a user uploads a new image, the user is waiting for a recommended policy. The two-stage approach allows the system to employ the first stage to classify the new image and find the candidate sets of images for the subsequent policy recommendation. As for the one-stage mining approach, it would not be able to

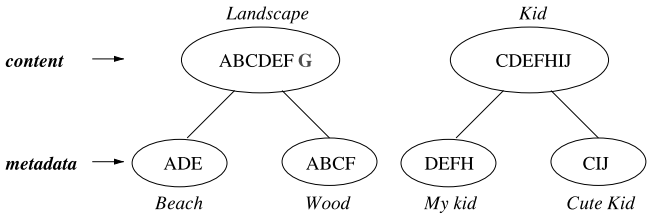

Fig. 2. Two-level Image classification.

locate the right class of the new image because its classification criteria needs both image features and policies whereas the policies of the new image are not available yet. Moreover, combining both image features and policies into a single classifier would lead to a system which is very dependent to the specific syntax of the policy. If a change in the supported policies were to be introduced, the whole learning model would need to change.

4.1 Image Classification

To obtain groups of images that may be associated with similar privacy preferences, we propose a hierarchical image classification which classifies images first based on their contents and then refine each category into subcategories based on their metadata. Images that do not have metadata will be grouped only by content. Such a hierarchical classification gives a higher priority to image content and minimizes the influence of missing tags. Note that it is possible that some images are included in multiple categories as long as they contain the typical content features or metadata of those categories.

Moreover, Fig. 2 shows an example of image classification for 10 images named as A, B, C, D, E, F, G, H, I, J, respectively. The content-based classification creates two categories: "landscape" and "kid". Images C, D, E and F are included in both categories as they show kids playing outdoor which satisfy the two themes: "landscape" and "kid". These two categories are further divided into subcategories based on tags associated with the images. As a result, we obtain two subcategories under each theme respectively. Notice that image G is not shown in any subcategory as it does not have any tag; image A shows up in both subcategories because it has tags indicating both "beach" and "wood".

4.1.1 Content-Based Classification

Our approach to content-based classification is based on an efficient and yet accurate image similarity approach.

Specifically, our classification algorithm compares image signatures defined based on quantified and sanitized version of Haar wavelet transformation. For each image, the wavelet transform encodes frequency and spatial information related to image color, size, invariant transform, shape, texture, symmetry, etc. Then, a small number of coefficients are selected to form the signature of the image. The content similarity among images is then determined by the distance among their image signatures.

Our selected similarity criteria include texture, symmetry, shape (radial symmetry and phase congruency [26]), and SIFT [25]. We also account for color and size. We set the system to start from five generic image classes: (a) explicit (e.g., nudity, violence, drinking etc), (b) adults, (c) kids, (d) scenery (e.g., beach, mountains), (e) animals. As a preprocessing step, we populate the five baseline classes by manually assigning to each class a number of images crawled from Google images, resulting in about 1,000 images per class. Having a large image data set beforehand reduces the chance of misclassification. Then, we generate signatures of all the images and store them in the database.

Upon adjusting the settings of our content classifier, we conducted some preliminary test to evaluate its accuracy. Precisely, we tested our classifier it against a ground-truth data set, Image-net.org [17]. In Image-net, over 10 million images are collected and classified according to the wordnet structure. For each image class, we use the first half set of images as the training data set and classify the next 800 images. The classification result was recorded as correct if the synset's main search term or the direct hypernym is returned as a class. The average accuracy of our classifier is above 94 percent.

Having verified the accuracy of the classifier, we now discuss how it is used in the context of the A3P core. When a user uploads an image, it is handled as an input query image. The signature of the newly uploaded image is compared with the signatures of images in the current image database. To determine the class of the uploaded image, we find its first m closest matches. The class of the uploaded image is then calculated as the class to which majority of the m images belong. If no predominant class is found, a new class is created for the image. Later on, if the predicted policy for this new image turns out correct, the image will be inserted into the corresponding image category in our image database, to help refine future policy prediction. In our current prototype, m is set to 25 which is obtained using a small training data set.

4.1.2 Metadata-Based Classification

The metadata-based classification groups images into subcategories under aforementioned baseline categories. The process consists of three main steps.

The first step is to extract keywords from the metadata associated with an image. The metadata considered in our work are tags, captions, and comments. We identify all the nouns, verbs and adjectives in the metadata and store them as metadata vectors $\tau_{noun} = \{t_1, t_2, \ldots, t_i\}$, $\tau_{verb} = \{t_1, t_2, \ldots, t_j\}$ and $\tau_{adj} = \{t_1, t_2, \ldots, t_k\}$, where i, j and k are the total number of nouns, verbs and adjectives respectively.

The second step is to derive a representative hypernym (denoted as h) from each metadata vector. We first retrieve the hypernym for each t_i in a metadata vector based on the Wordnet classification [39] and obtain a list of hypernym $\eta = \{(v_1, f_1), (v_2, f_2), \ldots\}$, where v denotes hypernym and f denotes its frequency. For example, consider a metadata vector $\tau = \{$"cousin","first steps", "baby boy"$\}$. We find that "cousin" and "baby boy" have the same hypernym "kid", and "first steps" has a hypernym "initiative". Correspondingly, we obtain the hypernym list $\eta = \{$(kid, 2), (initiative, 1)$\}$. In this list, we select the hypernym with the highest frequency to be the representative hypernym, e.g., "kid". In case that there are more than one hypernyms with the same frequency, we consider the hypernym closest to

the most relevant baseline class to be the representative hypernym. For example, if we have a hypernym list $\eta = \{(\text{kid}, 2), (\text{cousin}, 2), (\text{initiative}, 1)\}$, we will select "kid" to be the representative hypernym since it is closest to the baseline class "kids".

The third step is to find a subcategory that an image belongs to. This is an incremental procedure. At the beginning, the first image forms a subcategory as itself and the representative hypernyms of the image becomes the subcategory's representative hypernyms. Then, we compute the distance between representative hypernyms of a new incoming image and each existing subcategory. Given an image, let h_n, h_a and h_v denote its representative hypernyms in the metadata vectors corresponding to nouns, adjectives and verbs, respectively. For a subcategory c, Let h_n^c, h_a^c and h_v^c denotes its representative hynpernyms of nouns, adjectives and verbs, respectively. The distance between the image and the subcategory is computed as a weighted sum of the edit distance [38] between corresponding pair of representative hypernyms as shown in Equation (1), where w denotes the weight and D denotes the edit distance,

$$Dist_m = w_n \cdot D\big(h_n, h_n^c\big) + w_a \cdot D\big(h_a, h_a^c\big) + w_v \cdot D\big(h_v, h_v^c\big). \tag{1}$$

Note that $w_n + w_a + w_v = 1$, and $w_n > w_a > w_v$. In Equation (1), we give the highest weight to the hypernyms of the nouns because nouns are closest to the baseline classes. We consider the hypenyms of the adjectives as secondly important as the adjectives can help refine the baseline criteria. Finally, we consider the hypernymms of the verbs. By default, $w_n = 0.5$, $w_a = 0.3$ and $w_v = 0.2$.

Next we check if the closest subcategory has the distance value smaller than a threshold ϵ. If so, the new image will be included in to the subcategory and we update the representative hypernyms of the subcategory by keeping the hypernyms with the highest frequency. Otherwise, a new subcategory will be constructed for this image.

4.2 Adaptive Policy Prediction

The policy prediction algorithm provides a predicted policy of a newly uploaded image to the user for his/her reference. More importantly, the predicted policy will reflect the possible changes of a user's privacy concerns. The prediction process consists of three main phases: (i) policy normalization; (ii) policy mining; and (iii) policy prediction.

The policy normalization is a simple decomposition process to convert a user policy into a set of atomic rules in which the data (D) component is a single-element set.

4.2.1 Policy Mining

We propose a *hierarchical mining* approach for policy mining. Our approach leverages association rule mining techniques to discover popular patterns in policies. Policy mining is carried out within the same category of the new image because images in the same category are more likely under the similar level of privacy protection. The basic idea of the hierarchical mining is to follow a natural order in which a user defines a policy. Given an image, a user usually first decides who can access the image, then thinks about what specific access rights (e.g., view only or download) should be given, and finally refine the access conditions such as setting the expiration date. Correspondingly, the hierarchical mining first look for popular subjects defined by the user, then look for popular actions in the policies containing the popular subjects, and finally for popular conditions in the policies containing both popular subjects and conditions.

TABLE 1
Example of Subject Component

PolicyID	family	friend	coworker	others
P_2	0	0	1	0
P_5	1	1	0	0
P_9	1	1	0	0
P_{13}	1	1	0	0
P_{18}	0	1	1	0
P_{22}	1	0	0	0

- Step 1: In the same category of the new image, conduct association rule mining on the subject component of polices. Let S_1, S_2, …, denote the subjects occurring in policies. Each resultant rule is an implication of the form $X \Rightarrow Y$, where X, $Y \subseteq \{S_1, S_2, \ldots, \}$, and $X \cap Y = \emptyset$. Among the obtained rules, we select the best rules according to one of the interestingness measures, i.e., the generality of the rule, defined using support and confidence as introduced in [16]. The selected rules indicate the most popular subjects (i.e., single subject) or subject combinations (i.e., multiple subjects) in policies. In the subsequent steps, we consider policies which contain at least one subject in the selected rules. For clarity, we denote the set of such policies as Γ_i^{sub} corresponding to a selected rule R_i^{sub}.

Example 2. Suppose that there are six images in the same category of the newly uploaded image "park.jpg" and the corresponding policies are P_2, P_5, P_9, P_{13}, P_{18} and P_{22}.

Table 1 shows what subjects are mentioned in each policy. Mining data in Table 1 may return a best association rule like R_1^{sub}: $\{family\} \Rightarrow \{friend\}$, meaning that when the user specifies a policy for his family members, he tends to grant the same access right to his friends. In other words, $\{family, friend\}$ is a popular combination appearing in policies. According to R_1^{sub}, P_2 will be removed for further consideration since it does not contain any subject in R_1^{sub}.

- Step 2: In each policy set Γ_i^{sub}, we now conduct association rule mining on the action component. The result will be a set of association rules in the form of $X \Rightarrow Y$, where X, $Y \subseteq \{open, comment, tag, download\}$, and $X \cap Y = \emptyset$. Similar to the first step, we will select the best rules according to the generality interestingness. This time, the selected rules indicate the most popular combination of actions in policies with respect to each particular subject or subject combination. Policies which do not contain any action included in the selected rules will be removed. Given a selected rule R_j^{act},

TABLE 2
Example of Action Component

PolicyID	view-only	comment	tag	download
P_5	0	1	1	0
P_9	1	0	0	0
P_{13}	0	1	1	0
P_{18}	0	1	1	1
P_{22}	0	1	1	1

TABLE 3
Example of Condition Component

PolicyID	age	location	time	affiliation
P_5	1	1	1	0
P_{13}	1	0	1	0
P_{18}	1	0	1	0
P_{22}	0	0	0	1

we denote the set of remaining policies as Γ_j^{act}, and note that $\Gamma_j^{act} \subseteq \Gamma_i^{sub}$.

Example 3. Let us consider the remaining policies from Example 2. Table 2 shows the action components in these policies (actions "comment", "tag" and "download" imply the "view" action).

After mining the action component, we may obtain association rules as follows:

R_1^{act}: $\{tag\} \Rightarrow \{comment\}$

R_2^{act}: $\{download\} \Rightarrow \{comment\}$

R_1^{act} means that when the user allows someone to tag an image, he usually also allows the person to comment on the image. R_2^{act} means that if one has the "download" right of an image, he/she is most likely to also have the comment right.

Suppose that the best rule is R_1^{act} according to the interestingness measure. Then, policy P_9 will be removed.

- Step 3: We proceed to mine the condition component in each policy set Γ_j^{act}. Let $attr_1$, $attr_2$, ..., $attr_n$ denote the distinct attributes in the condition component of the policies in Γ_j^{act}. The association rules are in the same format of $X \Rightarrow Y$ but with X, $Y \subseteq \{attr_1, attr_2, \ldots, attr_n\}$. Once the rules are obtained, we again select the best rules using the generality interestingness measure. The selected rules give us a set of attributes which often appear in policies. Similarly, we denote the policies containing at least one attribute in the selected rule R_k^{con} as Γ_k^{con} and $\Gamma_k^{con} \subseteq \Gamma_j^{act}$.

  The next task is to determine the actual condition of these attributes. Specifically, in each Γ_k^{con}, we will choose the most frequent conditions for the selected attributes.

Example 4. Let us continue with Example 3. Table 3 lists attributes occurring in the condition component of the remaining policies.

The best association rule may be: R_1^{con}: $\{age\} \Rightarrow \{time\}$. It indicates that this user usually mentions age and time together in policy conditions. Consequently, policy P_{22} will be removed. Suppose that the majority of the policies (both P_5 and P_{13}) specify that people with age older than 18 will be granted access right before year 2012. Then, these conditions will be considered for generating candidate policies in the following Step 4.

- Step 4: This step is to generate candidate policies.

  Given $\Gamma_k^{con} \subseteq \Gamma_j^{act} \subseteq \Gamma_i^{sub}$, we consider each corresponding series of best rules: $R_{k_x}^{con}$, $R_{j_y}^{act}$ and $R_{i_z}^{sub}$. Candidate policies are required to possess all elements in $R_{k_x}^{con}$, $R_{j_y}^{act}$ and $R_{i_z}^{sub}$. Note that candidate policies may be different from the policies as result of Step 3. This is because Step 3 will keep policies as long as they have one of the attributes in the selected rules.

Example 5. From Example 2, 3 and 4, we obtained the following set of best association rules:

R_1^{sub}: $\{family\} \Rightarrow \{friend\}$

R_1^{act}: $\{tag\} \Rightarrow \{comment\}$

R_1^{con}: $\{age\} \Rightarrow \{time\}$

For the new image park.jpg, one candidate policy could be: P_{can}: [{family,friend}, {park.jpg}, {comment, tag}, ($age > 18 \wedge$time< 2012)]

4.2.2 Policy Prediction

The policy mining phase may generate several candidate policies while the goal of our system is to return the most promising one to the user. Thus, we present an approach to choose the best candidate policy that follows the user's privacy tendency.

To model the user's privacy tendency, we define a notion of *strictness level*. The strictness level is a quantitative metric that describes how "strict" a policy is. In particular, a strictness level L is an integer with minimum value in zero, wherein the lower the value, the higher the strictness level. It is generated by two metrics: major level (denoted as l) and coverage rate (α), where l is determined by the combination of subject and action in a policy, and α is determined by the system using the condition component. l is obtained via Table 4. In Table 4, all combinations of common subject and common actions are enumerated and assigned an integer value according to the strictness of the corresponding subjects and actions. For example, "view" action is considered more restricted than "tag" action. Given a policy, its l value can be looked up from the table by matching its subject and action. If the policy has multiple subjects or actions and results in multiple l values, we will consider the lowest one. It is worth noting that the table is automatically generated by the system but can be modified by users according to their needs.

Then, we introduce the computation of the coverage rate α which is designed to provide fine-grained strictness level. α is a value ranging from 0 to 1 and it will just adjust but not dominate the previously obtained major level. In particular, we define α as the percentage of people in the specified subject category who satisfy the condition in the policy. For example, a user has five family members documented in the system and two of them are kids. When he specifies a policy

TABLE 4
Major Level Look-Up Table

Major Level	Subject	Action
0	family	view
1	family	comment
2	family	tag
3	family	download
4	friend	view
5	friend	comment
6	friend	tag
7	friend	download
8	coworker	view
9	coworker	comment
10	coworker	tag
11	coworker	download
12	stranger	view
13	stranger	comment
14	stranger	tag
15	stranger	download

with the condition $age > 18$, only three family members will satisfy this condition. The corresponding α is then $3/5 = 0.6$. The larger the value of α, the more people are allowed to access the image and the policy is less restricted. Therefore, we subtract (1-α) from l to obtain the final strictness level as shown in Equation (2):

$$L = l - (1 - \alpha). \tag{2}$$

Example 6. Consider the candidate policy P_{can} in Example 5. It has two subjects $\{family, friend\}$ as well as two actions $\{comment, tag\}$. By looking up Table 4, we find that the combination of "friend–tag" yields the lowest major level, i.e., 6. Suppose that the obtained α is 0.3 after evaluating the condition against available user profiles. The final strictness level for P_{can} is P_{can}6-(1-0.3) = 5.3.

After we compute the strictness levels of all candidate policies, we now need to determine which strictness level fits best to the user's privacy trend. For this purpose, we propose the following approach. We keep monitoring the average strictness level of existing policies in each category of images. The average strictness level is defined as follows:

$$L_{avg} = \frac{\sum_{i=1}^{N_p} L_{p_i}}{N_p}, \tag{3}$$

where L_{p_i} denote the strictness level of policy P_i, and N_p is the total number of policies that satisfy $|L_{p_i} - Lavg| \leq \xi$. Notice that the average strictness level is computed by excluding outlier policies. This is because in some situations, users may define special policies which have a very different strictness level from most of others, either much more strict or much more loose. Considering such outliers into the average strictness level calculation would not represent the average case properly. Therefore, when a policy is inserted, we first compare its strictness level with current average strictness level. If the difference is more than a threshold (ξ), we put the policy in the outlier group. In the experiments, we set ξ to 4 because each role of the policy subject has four different strictness levels as shown in Table 4. The change on the policy preferences being more than four is considered prominent as it exceeds one quarter of the maximum strictness level.

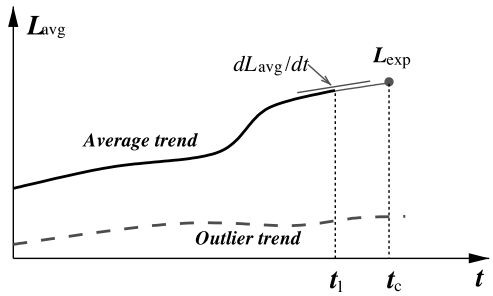

Fig. 3. Average strictness level curve.

As time evolves, the average strictness levels in each category form a curve as shown in Fig. 3, where values of strictness levels are interpolated in-between any consecutive policy updates. Similarly, the outlier policies may form their own curves as denoted in the figure.

Let t_l denote the last timestamp at which a policy is input to the system, and t_c denote the current timestamp when a new image is uploaded. We estimate the expected strictness level L_{exp} for the new image based on the derivative of the curve of the average strictness level at t_l. The derivative can be computed using Secant method [19]. In a summary, L_{exp} is computed by Equation (4):

$$L_{exp} = L_{avg}(t_l) + (t_c - t_l) \cdot \frac{dL_{avg}}{dt}(t_l). \tag{4}$$

We compare the strictness level of the candidate policies with L_{exp} of the average trend, and select the policy which has the closet value to L_{exp}. When there is more than one policy with strictness levels within the same distance to L_{exp}, we will conservatively choose the one with the lowest value, i.e., the more restrictive one. Once the user accepted or revised the recommended policy, the new policy will be added to the user's policy repository.

It is worth noting that the outlier trend may become the average trend at certain point as time passes, and during the transitional period, the policy prediction may not be very accurate. If a user suddenly changes his/her privacy strictness level to a much higher or much lower level, the prediction error of our approach for this single change will be high since we will treat this change as an outlier. If this change is the new preference of this user, this change will be identified when the number of images associated with this new privacy preference is larger than the number of the images associated with the average trend. At that point, the two curves will switch their roles.

5 A3P-Social

The A3P-social employs a multi-criteria inference mechanism that generates representative policies by leveraging key information related to the user's social context and his general attitude toward privacy. As mentioned earlier, A3P-social will be invoked by the A3P-core in two scenarios. One is when the user is a newbie of a site, and does not have enough images stored for the A3P-core to infer meaningful and customized policies. The other is when the system notices significant changes of privacy trend in the

user's social circle, which may be of interest for the user to possibly adjust his/her privacy settings accordingly.

In what follows, we first present the types of social context considered by A3P-Social, and then present the policy recommendation process.

5.1 Modeling Social Context

We observe that users with similar background tend to have similar privacy concerns, as seen in previous research studies and also confirmed by our collected data.

This observation inspires us to develop a social context modeling algorithm that can capture the common social elements of users and identify communities formed by the users with similar privacy concerns. The identified communities who have a rich set of images can then serve as the base of subsequent policy recommendation. The social context modeling algorithm consists of two major steps. The first step is to identify and formalize potentially important factors that may be informative of one's privacy settings. The second step is to group users based on the identified factors.

First, we model each user's social context as a list of attributes: $\{sc_1, sc_2, \ldots, sc_n\}$, where sc_i denote a social context attribute, and n is the total number of distinct attributes in the social networking site. These social context attributes are extracted from users' profiles.

Besides basic elements in users' profiles, many social sites also allow users to group their contacts based on relationships (e.g., friends, family members). If such grouping functionality is available, we will consider its influence on privacy settings too. In a social site, some users may only have their family members as contacts, while some users may have contacts including different kinds of people that they met offline or on the Internet. The distribution of contacts may shed light on the user's behavior of privacy settings. We assume that users who mainly share images among family members may not want to disclose personal information publicly, while users having a large group of friends may be willing to share more images with a larger audience [18]. Formally, we model the ratio of each type of relationship among all contacts of a user as *social connection*. Let $R_1, \ldots, R_n$ denote the n types of relationships observed among all users. Let $N^u_{R_i}$ denote the number of user U's contacts belonging to relationship type R_i. The connection distribution (denoted as $Conn$) is represented as below:

$$Conn : \left\{ \frac{N^u_{R_1}}{\sum_{i=1}^{n} N^u_{R_i}}, \ldots, \frac{N^u_{R_n}}{\sum_{i=1}^{n} N^u_{R_n}} \right\}.$$

For example, suppose that there are four types of relationships being used by users in the system: R_1 ="family", R_2 = "colleague", R_3 = "friend", R_4 = "others". Bob has 20 contacts, among which he has 10 family members, five colleagues, and five friends. His social connection is represented as $\{\frac{10}{20}, \frac{5}{20}, \frac{5}{20}, \frac{0}{20}\}$. It is worth noting that, the number of social context attributes may grow when more rich information is collected by social networking sites in the future, and our algorithm is dynamic and capable of dealing with any number of attributes being considered.

The second step is to identify groups of users who have similar social context and privacy preference. Regarding social context, it rarely happens that users share the same values of all social context attributes. More common cases are that a group of users have common values for a subset of social context attributes. Such subset can be different for different groups of users, which makes the user grouping a challenging task. We illustrate the scenario using the following example. For simplicity of illustration, we take a smaller set of attributes to be considered.

Example 7. Suppose that there are five users u_1, u_2, ..., u_5 in a social networking site. Each of them is associated with five social context attributes: gender, hobbies, occupation, location and social connection.

u_1: [Female, movie, accountant, NY, {0.6, 0.1, 0.2, 0.1}]
u_2: [Female, movie, teacher, IL, {0.7, 0.1, 0.1, 0.1}]
u_3: [Male, ski, student, CO, {0.3, 0.1, 0.5, 0.1}]
u_4: [Male, ski, student, KS, {0.6, 0.15, 0.15, 0.1}]
u_5: [Male, ski, student, MO, {0.2, 0.1, 0.6, 0.1}]

From the users' profile and social connection, two natural social groups can be formed.

$G_1 = \{u_1, u_2\}$, since they are both female who love movies and frequently share data with family members.

$G_2 = \{u_3, u_4, u_5\}$, since they are all male students who love sports.

In the above example, we observe that the first social group is formed based on attributes: gender, hobbies and social connection, while the second social group is formed based on a different set of attributes which are gender, hobbies and occupation. To identify such dynamic social groups, we employ an a priori [2]-based data mining Algorithm. Unlike the original a priori which requires exact matches of items in different transactions, we allow a fuzzy mapping. This is important for matching the social connection attribute which would be just slightly different in the same social group as shown in Example 7. Therefore, we define the matching of social connection attribute as the values of this attribute within a small threshold. Definition 2 describes the social groups extracted by our algorithm.

Definition 2 (Social Group). *Let U be the set of all users, and SC be the universe of social context attributes, SC=$\{sc_1, sc_2, \ldots\}$. Let G be a subset of users U, i.e., $G \subseteq U$; and let C be the subset of SC, $C \subseteq SC$. We say G is a social group if it satisfies all the following conditions:*

- *For any user $u_i, u_j \in G$, they have matching values for each attribute in C but not any superset of C;*
- *The number of users in G is greater than threshold k_u;*
- *The number of attributes in C is greater than k_c.*

k_u and k_c are thresholds for determining the size of a social group, which are set to and respectively by default.

The algorithm proceeds as follows. We first group users who have only one matching attribute, and keep the sets that contain more than k_u users. We call such sets as 1-attribute social group and denote it as $G1$. Then, we join every pair of 1-attribute social groups. For example, given two 1-attribute social groups $G1_1$ and $G1_2$, let $G2 = G1_1 \cap G1_2$. If $|G2| > k_u$, then $G2$ is a 2-attribute social group whereby users in $G2$ have two common attributes. Next, we join every pair of 2-attribute social groups that have at least one attribute in common. Let $G2_1$ and $G2_2$ be two 2-attribute social groups respectively and $C2_1 = \{a_1, a_2\}$ and $C2_2 = \{a_2, a_3\}$ be their

attribute sets respectively. Observe that $|C2_1$ and $C2_2$ have one common attribute, and $|C2_1 \cup C2_2| = 3$. If $|G2_1 \cap G2_2| > k_u$, then $G3 = G2_1 \cap G2_2$ is a 3-attribute social group which means the users have three attributes in common. Similarly, by joining every pair of 3-attribute social groups that have two common attributes, we can obtain 4-attribute social groups. This process keeps going until no more social groups can be produced. During the process, if a k-attribute group cannot be used to produce $(k+1)$ group, this k-attribute group is included in the final results.

The obtained social groups have not taken into account privacy preferences yet. It is certainly possible users within the same social group maintain various privacy preferences. In order to tie social groups to privacy preferences, we further divide the social groups into sub-groups according to the closeness of their privacy preferences. In particular, we sort the users in the same social group in an ascending order of their privacy strictness levels. Then, starting from the user (say u_i) with the minimum strictness level (L_i), we scan the sorted list and include users whose strictness levels are no more than $L_i + \xi$. After one subgroup is formed, we remove the users in the subgroup from the sorted list. If the sorted list is not empty, we start the subgroup formation again in the same way until all users have been grouped. The following example illustrates the steps.

Example 8. Suppose that there are six users in a social group, and their privacy strictness levels are included in the parenthesis right after their user ID: $U_1(14)$, $U_2(2)$, $U_3(4)$, $U_4(11)$, $U_5(3)$, $U_6(12)$. We form sub-groups as follows.

The six users are first sorted according to the strictness level, and the result is:

$U_2(2), U_5(3), U_3(4), U_4(11), U_6(12), U_1(14)$.

The first three users in the sorted list have strictness levels with difference no more than 4, and form one subgroup. Similarly, the remaining three form another subgroup. The final sub-groups are listed below, which contain users with both similar social context and privacy preferences:

SG_1: $\{U_2, U_5, U_3\}$,
SG_2: $\{U_4, U_6, U_1\}$.

5.2 Identifying Social Group

We now introduce the policy recommendation process based on the social groups obtained from the previous step.

Suppose that a user U uploaded a new image and the A3P-core invoked the A3P-social for policy recommendation. The A3P-social will find the social group which is most similar to user U and then choose the representative user in the social group along with his images to be sent to the A3P-Core policy prediction module to generate the recommended policy for user U. Given that the number of users in social network may be huge and that users may join a large number of social groups, it would be very time consuming to compare the new user's social context attributes against the frequent pattern of each social group. In order to speed up the group identification process and ensure reasonable response time, we leverage the inverted file structure [31] to organize the social group information.

The inverted file maps keywords (values of social context attribute) occurring in the frequent patterns to the social groups that contain the keywords. Specifically, we first sort the keywords (except the social connection) in the frequent patterns in an alphabetical order. Each keyword is associated with a link list which stores social group ID and pointers to the detailed information of the social group. The following example illustrates the detailed structure.

Example 9. Suppose that there are three social groups G_1, G_2, G_3 which are formed based on the following frequent keywords.

G_1: {female, movie, {0.6, 0.1, 0.2, 0.1}}
G_2: {male, ski, student}
G_3: {male, movie, IL }

We select the frequent attribute values except the social connection and build an inverted file as follows.

female: $\{G_1\}$
IL: $\{G_3\}$
hiking: $\{G_3\}$
male: $\{G_2, G_3\}$
movie: $\{G_1, G_3\}$
student: $\{G_2\}$

Next, given a new user, we search his/her attribute values in the inverted file and obtain a set of candidate social groups. We also count the number of occurrence of the candidate groups during the search. We select the candidate group with the highest occurrence as the social group for the new user. For example, given a user whose social context attributes are: {female, movie, teacher, NY, {0.65, 0.1, 0.15, 0.1}}, we find that only the keywords "female" and "movie" appear in the inverted file. The social group related to "female" is G_1, and the social groups related to "movie" are G_1 and G_3. Observe that G_1 occurs twice in the search and G_2 only once. That means the new user has more matching keywords with G_1 than G_2 and other social groups, and hence G_1 is a better group for the new user.

In the identified social group, we further examine its subgroups by comparing the strictness levels of the sub-groups with the new user's preferred privacy strictness level if provided. We select the sub-group whose strictness level matches the new user's privacy requirements best. If the new user did not specify privacy preference, we select the sub-group with the largest members. Then, in this selected sub-group, we look for the user who is most similar to the new user. We just need to compare the new user's and the group members' remaining attributes that are not included in the frequent pattern. The selected user and his/her images and policies are sent to the A3P-Core module to generate the recommended policy for the new user.

Finally, we update the social group information by including the new user as a probational member. The probational member will not be chosen by A3P-Social module to until he/she uploaded sufficient images and becomes a regular member.

6 Experimental Evaluation

We evaluate the effectiveness of our A3P system in terms of the policy prediction accuracy and user acceptability. The A3P was implemented as a Java file embedded in an

open source content management site, deployed using an Apache server.

6.1 Experimental Settings

We conduct and collect data sets for two types of experiments: survey-based study and direct user evaluation.

Survey-based study and data collection. We collected two sets of actual user-specified policies to be used as ground truth for our evaluation.

Data collection 1. This study involved 88 participants (48 female and 40 males) who were recruited from a large US university community (staff, students, and the community at large). Their average age is 26.3 years old (Range: 18-39). The participants completed at least 90 percent of the questionnaire consisting of two parts. The first part contains questions related to one's background information and online privacy practices and the second part is to collect user-specified policies.

In the first part of the questionnaire, the participants were asked to indicate any social networks they were a part of (98 percent indicated Facebook and 37 percent also indicated others like Myspace). In terms of usage frequency, 95 percent of the respondents accessed social network sites at least once a week, with 76 percent of reporting that they were daily users.

We also asked participants if they have had concerns about their privacy due to shared images. Over 51 percent of the participants indicated that they had privacy concerns. Users also reported that image content is an important factor when determining privacy settings for an image with 87 percent of people agreeing or strongly agreeing with the statement "When I set privacy settings for a certain image I usually think about the content of the image", and over 91 percent of users agreeing or strongly agreeing with the statement "The content of an image determines whether I upload the image to a social network site." Surprisingly, however, many users indicated that they never changed privacy settings for images (38 percent) or changed their settings only 1 or 2 times (36 percent) since joining the social network. There seems to be a clear disconnect between users privacy inclinations and their practice of setting privacy policies. The possible reason could be "Changing privacy settings for every image uploaded on a social site can be very time consuming", as strong agreed or agreed by 70 percent of users.

In the second part of the questionnaire, we presented each user 30 images selected by us. For each image, we asked the user to input the privacy settings by assuming these photos as his/her own images. We collected around 3,000 policies.

Data collection 2. The second study involved 67 new users recruited using Amazon Mechanical Turk. Each user was given a distinct set of up to 130 images taken from the Picalert project data set [44] including Flickr images on various subjects and different privacy sensitivity. On average, each user labeled 57 images with their policies and added two to three tags each. These participants consist of 36 male and 31 female with age ranging in 34-39. Similar questions on privacy concerns and practices were asked as seen from the previous data collection task. Ninety-seven percent of the respondents declare to be social network users, and 78 percent of them access their sites daily. About half of the respondents (45 percent) maintain "default" privacy settings in their profile, and only 6 percent have a private profile. Fifty-eight percent of the users declare to be part of one or more social groups. In addition, this population appears to be mostly of data consumers, in that only 21 percent of the users declare to have more than 150 images posted (28 percent have less than 50 images on their profile).

In this experiment, we slightly changed the policy format to be inline with the policies adopted by Flickr. Specifically, each user was asked three separate questions for every image: (i) who can view the image? (ii) who can comment? and (iii) who can add notes, tags, and download it?. For each question, the user may choose one among the following options: only you, family only, friends only, social network contacts, and everyone. Note that our policy mining algorithm can easily adapt to different formats of policies. Out of the 67 users, we kept data for 56 of them, as the remaining ones generated poor quality data (e.g., data was incomplete, all the policies entered were the same for all images or many tags were missing). In total, the users produced 2,004 policies for 2,004 distinct images. Also, users added about 8,600 tags to the images (avg. 2.5/image).

Overall, the two data sets allow us to collect data from a diverse group of social network users with varying levels of reported usage. A common theme across all users is that the visual content of an image is an important aspect in determining both whether to upload the image to a social network site as well as its specific visibility within the site. However, while users expressed general privacy concerns regarding shared images, many users reported that they did not change privacy settings for images (about 50 percent of them, overall). These seemingly conflicting responses highlight the need for tools to help users ensure that privacy settings are meeting expectations.

Direct user evaluation. We invited another 41 people to use our A3P system. The goal of this experiment is to assess our system's acceptability, i.e., whether users would consider the predicted policies reasonable, and inline with their overall preferences. We asked participants to input policies for a few images at first for training purposes. Three images from a given class are sufficient to bootstrap the algorithm. Next, participants enter privacy settings for a set of images that they would upload in their fictitious profile. Upon showing the image, privacy settings for it are suggested to the user. The participant has the option to accept the predicted policy as is, revise some components of it, or disagree with the predicted result and e re-enter preferred settings.

6.2 Large Scale Evaluation and Analysis

In this first round of tests, we used the two data sets collected through our survey to evaluate the accuracy of our recommended policies.

6.2.1 A3P-Core

Our first experiment compares A3P-core with alternative prediction approaches. In particular, we use a straw man solution as the baseline approach, whereby we sample at random a small set of image settings from the same user

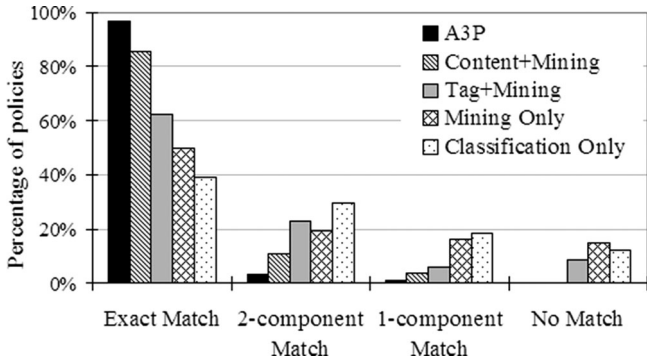

Fig. 4. A3P comparative performance.

and use them to determine a baseline setting (by counting the most frequent items). The baseline settings are applied to all images of the users. Further, we compare the A3P-core with two variants of itself, in order to evaluate the contribution of each component in the A3P-core made for privacy prediction. The first variant uses only content-based image classification followed by our policy mining algorithm, denoted as "Content+Mining". The second variant uses only tag classification followed by the policy mining, denoted as "Tag+Mining". All the algorithms were tested against the collected real user policies. Fig. 4 shows the percentage of predicted policies in four groups: "Exact Match" means a predicted policy is exactly the same as the real policy of the same image; "x-component Match" means a predicted policy and its corresponding real policy have x components (i.e., subject, action, condition) fully matched; "No match" simply means that the predicted policy is wrong for all components. As shown in the figure, each component of the A3P-core singularly contributes toward policy prediction, however, none of them individually equalizes the accuracy achieved by the A3P-core in its entirety. Specifically, A3P-core has 90 percent exact match and 0 no match. Moreover, pairwise comparisons were made between A3P-core, "Content+Mining, "Tag+Mining" and the baseline algorithm, corrected using a Bonferroni method [6]. Analyses indicate that A3P-core performed better than "Content+Mining" ($t(87) = 6.67,\ p < .001$), "Tag +Mining" ($t(87) = 8.25,\ p < .001$) and baseline ($t(87) = 15.452,\ p < .001$).

TABLE 5
Predictors of Performance -Stan B = Standardized B

Variable	B	Stan B	95% CI for B
Constant	20.864**		[17.001, 24.728]
Freq social network	0.105	0.022	[−2.051, 2.261]
Freq sharing pictures	0.003	0.002	[−.357, 0.363]
Freq changing privacy	0.043	0.050	[−.165,0.251]
Content of concern	1.407**	0.461	[.699, 2.116]
Privacy concern	0.263*	0.329	[0.072, 0.455]
Privacy takes time	0.106	0.071	[−.531, 0.742]
R^2		.231	
F		3.402*	

CI = confidence interval $^{*}p < .01$,$^{**}p < .001$.

TABLE 6
Results of A3P-Core on Picalert Data Set

Method	View	Comment	Tag, Notes, Download	Overall
A3P-core	92.48%	92.48 %	92.63 %	92.53 %
Propagation	66.12 %	66.825 %	68.64 %	66.84 %
Tag-Only	87.54%	87.03 %	86.64 %	87. 01 %

We complete this experiment on the second data set of over 2,000 images. The goal is to investigate whether the different population, and the heterogeneous set of images from the second data set influences the quality of the prediction. Also, this data set is characterized by a better meta-data, as manual inspection revealed that the user-entered tags are all completed, meaningful and with little jargon or use of stop words within them. For this experiment, we again used the straw man approach for comparison which consisted of replicating the latest generated policy by the user. This comparison is needed to remove the doubt that users of Mechanical turk may be completing crowd sourcing tasks in a automated fashion, without paying proper attention to each individual task. We also tested the quality achieved by A3P-core in case tags only were used, since the previous experiment showed that tags had little relevance for the prediction purpose. Results are reported in Table 6. As shown, A3P-core performed well, and showed an accuracy similar to the previous experiment (above 92.4 percent). We note that the accuracy per user ranged from 85 to 100 percent. The straw man approach performed very poorly, whereas the A3P-core on meta data only showed a drastic improvement compared to the previous tests. The accuracy is around 87 percent whereas Tag+Mining was only at 60 percent in the previous rounds of experiments. This is motivated by the better meta-data added by the participants.

6.2.2 Analysis of Users' Characteristics

We are also interested in examining whether our algorithm performs better for users with certain characteristics. Therefore, we study possible factors relevant to the performance of our algorithm. We used a least squares multiple regression analysis, regressing performance of the A3P-core to the following possible predictors:

- *Frequency of social network use* was measured on a frequency rating scale (1 = daily; 2 = weekly; 3 = monthly; 4 = rarely; 5 = never) with the item 'How often do you access Social Network Sites?'
- *Privacy settings take time* was measured on a Likert Scale (5-point rating scale, where 1 = strongly agree and 5 = strongly disagree) with the item 'Changing privacy settings for images uploaded on a social site can be very time consuming.'
- *Frequency of sharing pictures* was measured using three items ($\alpha = 0.69$) rated on a Likert scale.
- *Frequency of changing privacy settings* was measured using four items ($\alpha = 0.86$) rated on a Likert scale.

An example item is 'I have changed privacy settings for individual pictures.'

- *Content of concern* was measured using three items ($\alpha = 0.81$) rated on a Likert scale. An example item is 'The content of an image is of concern when determining the privacy level for an image.'
- *Privacy concern* was measured using four items ($\alpha = 0.76$) rated on a Likert scale. An example item is 'I have had concerns about my privacy due to shared images on social network sites.'

The model results are shown in Table 5. We can observe that the content of concern variable was the biggest predictor of performance of our algorithm (standardized $\beta = 0.461$, $p < 0.001$). This suggests the importance of content in determining the privacy level of uploaded images to social network sites. Privacy concern was also a significant predictor of performance (standardized $\beta = 0.329$, $p < 0.01$) with increased performance for those users who felt that images uploaded to social network sites allowed for exposure of personal information. Surprisingly, none of the other predictors were significantly related to performance of the A3P-core. We expected that frequency of sharing pictures and frequency of changing privacy settings would be significantly related to performance, but the results indicate that the frequency of social network use, frequency of uploading images and frequency of changing settings are not related to the performance our algorithm obtains with privacy settings predictions. This is a particularly useful result as it indicates that our algorithm will perform equally well for users who frequently use and share images on social networks as well as for users who may have limited access or limited information to share.

6.2.3 A3P Social

In the second round of experiments, we analyze the performance of the A3P-Social component by using the first set of data collection. For each user, we use the A3P-Social to predict policies and compare it with a base-line algorithm which does not consider social contexts but bases recommendation only on social groups that have similar privacy strictness level for same type of images. Using the base-line approach, we note that regardless of the individual privacy inclination of the users, the best accuracy is achieved in case of explicit images and images dominated by the appearance of children. In both cases, users maintain more consistent policies, and our algorithm is able to learn them effectively. The largest variability, and therefore worse results occur for images denoting scenery, where the error rate is 15.2 percent. Overall, the accuracy achieved by grouping users by strictness level is 86.4 percent.

With A3P-Social, we achieve a much higher accuracy, demonstrating that just simply considering privacy inclination is not enough, and that "social-context" truly matters. Precisely the overall accuracy of A3P-social is above 95 percent. For 88.6 percent of the users, all predicted policies are correct, and the number of missed policies is 33 (for over 2,600 predictions). Also, we note that in this case, there is no significant difference across image types. For completeness, we compared the performance of the A3P-Social with alternative, popular, recommendation methods: *Cosine* and *Pearson* similarity [34]. Cosine similarity is a measure of similarity between two vectors of an inner product space that measures the cosine of the angle between them. In our case, the vectors are the users' attributes defining their social profile. The algorithm using Cosine similarity scans all users profiles, computes Cosine similarity of the social contexts between the new user and the existing users. Then, it finds the top two users with the highest similarity score with the candidate user and feeds the associated images to the remaining functions in the A3P-core.

Pearsons similarity instead measures how highly correlated are two variables, and is usually used to correlate users' ratings on recommended products. To adapt, we replaced the users rating from the Pearson similarity with self-given privacy ratings, that is, we tested similarity based on how users rate their own privacy inclinations.

The data we use for this assumption is the response to three privacy-related questions users provide on their pre-session survey during data collection (the questions are adapted from the well-known privacy-index measures from Westin). Accordingly, we use Pearson similarity to find other users who are similar to this new user. With Pearson, we obtain an accuracy of 81.4 percent. We note however that 2-components accuracy is only about 1.77 percent of the missed policies, and even less 1-component. A similar result is obtained with Cosine similarity, where we achieved 82.56 percent accuracy, with again less than 2 percent accuracy for 2-components match and about 0.05 percent for 1-component.

In sum, A3P social appears to be always superior to other methods. Note however that we cannot use A3P-social alone without A3P-core since the A3P-social does not factor in the evolution of an individual's privacy preferences. Also A3P-social is more costly to be executed than A3P-core since the A3P-social analyzes information from a community rather than a single user.

TABLE 7
Result of Direct User Evaluation

Item Type	Count	Ratio
Total Policies	1025	
Exactly Matched Policies	944	92.1%
Policies with 1 error	67	6.4%
Policies with 2 errors	10	1.1%
Policies with 3 errors	4	0.4%

6.3 Direct User Evaluation

Table 7 reports the results for the direct user evaluation. Among a total of 1,025 predicted policies, we achieve over 92 percent accuracy ($SD = 0.047$), in that each participant rejected about two policies on average (1.98). The overall accuracy of the predicted policies in the direct user evaluation is significantly better than the performance in the off-line evaluation ($t(127) = 3.346, p < .01$). This demonstrates that users may not have a strong preference regarding privacy settings for individual pictures, and that a system like

A3P that can accurately predict preferences will lead to an acceptable level of privacy for users.

For mismatched policies, we further examined the type of error. We found that there were total 97 mismatched items (i.e., mismatched subjects, actions and conditions) in those policies. About 60 percent of the errors were due to false positive, which means the predicted policy contains more items than the actual policy. We also noticed that 82.7 percent of the mismatched policies have two components, the subject and action component, fully matched. The most common errors occur within the condition component as this component is the most flexible and can vary significantly if users want to add special constraints. Interestingly, the errors were reported mainly in the first three or four policies displayed to the user. This demonstrates the adaptive nature of our A3P system. Upon correcting mismatched policies, our system's accuracy increases. We also expect that with more user data and a longer execution of the A3P system, the prediction accuracy will be further increased, as the system adapts to users' privacy preferences.

7 Conclusion

We have proposed an Adaptive Privacy Policy Prediction (A3P) system that helps users automate the privacy policy settings for their uploaded images. The A3P system provides a comprehensive framework to infer privacy preferences based on the information available for a given user. We also effectively tackled the issue of cold-start, leveraging social context information. Our experimental study proves that our A3P is a practical tool that offers significant improvements over current approaches to privacy.

Acknowledgements

Squicciarini's work was partially funded by the US National Science Foundation (NSF-CNS-0831360) and a Google Research Award. Lin's work was funded by the US National Science Foundation (NSF-CNS-1250327).

References

[1] A. Acquisti and R. Gross, "Imagined communities: Awareness, information sharing, and privacy on the facebook," in *Proc. 6th Int. Conf. Privacy Enhancing Technol. Workshop*, 2006, pp. 36–58.

[2] R. Agrawal and R. Srikant,"Fast algorithms for mining association rules in large databases," in *Proc. 20th Int. Conf. Very Large Data Bases*, 1994, pp. 487–499.

[3] S. Ahern, D. Eckles, N. S. Good, S. King, M. Naaman, and R. Nair, "Over-exposed?: Privacy patterns and considerations in online and mobile photo sharing," in *Proc. Conf. Human Factors Comput. Syst.*, 2007, pp. 357–366.

[4] M. Ames and M. Naaman, "Why we tag: Motivations for annotation in mobile and online media," in *Proc. Conf. Human Factors Comput. Syst.*, 2007, pp. 971–980.

[5] A. Besmer and H. Lipford, "Tagged photos: Concerns, perceptions, and protections," in *Proc. 27th Int. Conf. Extended Abstracts Human Factors Comput. Syst.*, 2009, pp. 4585–4590.

[6] D. G. Altman and J. M. Bland ,"Multiple significance tests: The bonferroni method," *Brit. Med. J.*, vol. 310, no. 6973, 1995.

[7] J. Bonneau, J. Anderson, and L. Church, "Privacy suites: Shared privacy for social networks," in *Proc. Symp. Usable Privacy Security*, 2009.

[8] J. Bonneau, J. Anderson, and G. Danezis, "Prying data out of a social network," in *Proc. Int. Conf. Adv. Soc. Netw. Anal. Mining.*, 2009, pp.249–254.

[9] H.-M. Chen, M.-H. Chang, P.-C. Chang, M.-C. Tien, W. H. Hsu, and J.-L. Wu, "Sheepdog: Group and tag recommendation for flickr photos by automatic search-based learning," in *Proc. 16th ACM Int. Conf. Multimedia*, 2008, pp. 737–740.

[10] M. D. Choudhury, H. Sundaram, Y.-R. Lin, A. John, and D. D. Seligmann, "Connecting content to community in social media via image content, user tags and user communication," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2009, pp.1238–1241.

[11] L. Church, J. Anderson, J. Bonneau, and F. Stajano, "Privacy stories: Confidence on privacy behaviors through end user programming," in *Proc. 5th Symp. Usable Privacy Security*, 2009.

[12] R. da Silva Torres and A. Falcão, "Content-based image retrieval: Theory and applications," *Revista de Informática Teórica e Aplicada*, vol. 2, no. 13, pp. 161–185, 2006.

[13] R. Datta, D. Joshi, J. Li, and J. Wang, "Image retrieval: Ideas, influences, and trends of the new age," *ACM Comput. Surv.*, vol. 40, no. 2, p. 5, 2008.

[14] J. Deng, A. C. Berg, K. Li, and L. Fei-Fei, "What does classifying more than 10,000 image categories tell us?" in *Proc. 11th Eur. Conf. Comput. Vis.: Part V*, 2010, pp. 71–84. [Online]. Available: http://portal.acm.org/citation.cfm?id=1888150.1888157

[15] A. Kapadia, F. Adu-Oppong, C. K. Gardiner, and P. P. Tsang, "Social circles: Tackling privacy in social networks," in *Proc. Symp. Usable Privacy Security*, 2008.

[16] L. Geng and H. J. Hamilton, "Interestingness measures for data mining: A survey," *ACM Comput. Surv.*, vol. 38, no. 3, p. 9, 2006.

[17] Image-net data set. [Online]. Available: www.image-net.org, Dec. 2013.

[18] S. Jones and E. O'Neill, "Contextual dynamics of group-based sharing decisions," in *Proc. Conf. Human Factors Comput. Syst.*, 2011, pp. 1777–1786. [Online]. Available: http://doi.acm.org/10.1145/1978942.1979200

[19] A. Kaw and E. Kalu, *Numerical Methods with Applications: Abridged.*, Raleigh, North Carolina, USA: Lulu.com, 2010.

[20] 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 *Proc. ACM Annu. Conf. Human Factors Comput. Syst.*, 2012, pp. 377–386.

[21] K. Lerman, A. Plangprasopchok, and C. Wong, "Personalizing image search results on flickr," *CoRR*, vol. abs/0704.1676, 2007.

[22] H. Lipford, A. Besmer, and J. Watson, "Understanding privacy settings in facebook with an audience view," in *Proc. Conf. Usability, Psychol., Security*, 2008.

[23] D. Liu, X.-S. Hua, M. Wang, and H.-J. Zhang, "Retagging social images based on visual and semantic consistency," in *Proc. 19th ACM Int. Conf. World Wide Web*, 2010, pp.1149–1150.

[24] Y. Liu, K. P. Gummadi, B. Krishnamurthy, and A. Mislove, "Analyzing facebook privacy settings: User expectations vs. reality," in *Proc. ACM SIGCOMM Conf. Internet Meas. Conf.*, 2011, pp. 61–70.

[25] D. G. Lowe, (2004, Nov.). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* [Online]. *60(2)*, pp. 91–110. Available: http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94

[26] G. Loy and A. Zelinsky, "Fast radial symmetry for detecting points of interest," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 8, pp. 959–973, Aug. 2003.

[27] E. M. Maximilien, T. Grandison, T. Sun, D. Richardson, S. Guo, and K. Liu, "Privacy-as-a-service: Models, algorithms, and results on the Facebook platform," in *Proc. Web 2.0 Security Privacy Workshop*, 2009.

[28] A. Mazzia, K. LeFevre, and A. E.,, "The PViz comprehension tool for social network privacy settings," in *Proc. Symp. Usable Privacy Security*, 2012.

[29] M. Rabbath, P. Sandhaus, and S. Boll, "Analysing facebook features to support event detection for photo-based facebook applications," in *Proc. 2nd ACM Int. Conf. Multimedia Retrieval*, 2012, pp. 11:1–11:8.

[30] R. Ravichandran, M. Benisch, P. Kelley, and N. Sadeh, "Capturing social networking privacy preferences," in *Proc. Symp. Usable Privacy Security*, 2009.

[31] A. Singhal, "Modern information retrieval: A brief overview," *IEEE Data Eng. Bullet., Speicial Issue on Text Databases*, vol. 24, no. 4, pp. 35-43, Dec. 2001.

[32] A. C. Squicciarini, S. Sundareswaran, D. Lin, and J. Wede, "A3p: Adaptive policy prediction for shared images over popular content sharing sites," in *Proc. 22nd ACM Conf. Hypertext Hypermedia*, 2011, pp.261–270.

[33] K. Strater and H. Lipford, "Strategies and struggles with privacy in an online social networking community," in *Proc. Brit. Comput. Soc. Conf. Human-Comput. Interact.*, 2008, pp.111–119.
[34] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artif. Intell.*, vol. 2009, p. 4, 2009.
[35] X. Sun, H. Yao, R. Ji, and S. Liu, "Photo assessment based on computational visual attention model," in *Proc. 17th ACM Int. Conf. Multimedia*, 2009, pp. 541–544. [Online]. Available: http://doi.acm.org/10.1145/1631272.1631351
[36] H. Sundaram, L. Xie, M. De Choudhury, Y. Lin, and A. Natsev, "Multimedia semantics: Interactions between content and community," *Proc. IEEE*, vol. 100, no. 9, pp. 2737–2758, Sep. 2012.
[37] A. Vailaya, A. Jain, and H. J. Zhang, (1998). On image classification: City images vs. landscapes. *Pattern Recog.* [Online]. *31*(*12*), pp. 1921–1935. Available: http://www.sciencedirect.com/science/article/pii/S003132039800079X
[38] R. A. Wagner and M. J. Fischer, "The string-to-string correction problem," *J. ACM*, vol. 21, no. 1, pp. 168–173, 1974.
[39] Wordnet - A lexical database for the English language. [Online]. Available: http://wordnet.princeton.edu/
[40] C.-H. Yeh, Y.-C. Ho, B. A. Barsky, and M. Ouhyoung, "Personalized photograph ranking and selection system," in *Proc. Int. Conf. Multimedia*, 2010, pp. 211–220. [Online]. Available: http://doi.acm.org/10.1145/1873951.1873963
[41] C. A. Yeung, L. Kagal, N. Gibbins, and N. Shadbolt, "Providing access control to online photo albums based on tags and linked data," in *Proc. Soc. Semantic Web: Where Web 2.0 Meets Web 3.0 at the AAAI Symp.*, 2009, pp. 9–14.
[42] J. Yu, D. Joshi, and J. Luo, "Connecting people in photo-sharing sites by photo content and user annotations," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2009, pp.1464–1467.
[43] S. Zerr, S. Siersdorfer, J. Hare, and E. Demidova, "Privacy-aware image classification and search," in *Proc. 35th Int. ACM SIGIR Conf. Res. Develop. Inform. Retrieval*, 2012, pp. 35–44.
[44] S. Zerr, J. H. Stefan Siersdorfer, and E. Demidova, (2012). Picalert! data set. [Online]. Available: http://l3s.de/picalert/
[45] N. Zheng, Q. Li, S. Liao, and L. Zhang, "Which photo groups should I choose? A comparative study of recommendation algorithms in flickr," *J. Inform. Sci.*, vol. 36, pp. 733–750, Dec. 2010.
[46] J. Zhuang and S. C. H. Hoi, "Non-parametric kernel ranking approach for social image retrieval," in *Proc. ACM Int. Conf. Image Video Retrieval*, 2010, pp. 26–33. [Online]. Available: http://doi.acm.org/10.1145/1816041.1816047

Anna Cinzia Squicciarini received the PhD degree in computer science from the University of Milan, Italy, in 2006. She is an assistant professor in the College of Information Science and Technology at the Pennsylvania State University. During the years of 2006-2007, she was a post doctoral research associate at Purdue University. Her main interests include access control for distributed systems, privacy, security for Web 2.0 technologies, and grid computing. She is the author or co-author of more than 60 articles published in refereed journals, and in proceedings of international conferences and symposia. She is a member of the IEEE.

Dan Lin received the PhD degree in computer science from the National University of Singapore in 2007, and was a post doctoral research associate at Purdue University for two years. She is an assistant professor at the Missouri University of Science and Technology. Her main research interests include many areas in the fields of database systems, information security, cloud computing, and vehicular ad-hoc networks.

Smitha Sundareswaran received the bachelor's degree in electronics and communications engineering in 2005 from Jawaharlal Nehru Technological University, Hyderabad, India. She is currently working toward the PhD degree in the College of Information Sciences and Technology at the Pennsylvania State University. Her research interests include policy formulation and management for distributed computing architectures.

Josh Wede received the PhD degree in cognitive psychology from Purdue University in 2008. He is a senior lecturer in the Department of Psychology at the Pennsylvania State University. He studies visual information processing and attentional effects on neural representations.