The PViz Comprehension Tool for Social Network Privacy Settings

Alessandra Mazzia
Computer Science and Engineering
University of Michigan
2260 Hayward Ave.
Ann Arbor, MI 48109
amazzia@umich.edu

Kristen LeFevre
Computer Science and Engineering
University of Michigan
2260 Hayward Ave.
Ann Arbor, MI 48109
klefevre@umich.edu

Eytan Adar
School of Information
University of Michigan
105 South State St.
Ann Arbor, MI 48109
eadar@umich.edu

ABSTRACT

Users' mental models of privacy and visibility in social networks often involve subgroups within their local networks of friends. Many social networking sites have begun building interfaces to support grouping, like Facebook's lists and "Smart Lists," and Google+'s "Circles." However, existing policy comprehension tools, such as Facebook's *Audience View*, are not aligned with this mental model. In this paper, we introduce *PViz*, an interface and system that corresponds more directly with how users model groups and privacy policies applied to their networks. PViz allows the user to understand the visibility of her profile according to automatically-constructed, natural sub-groupings of friends, and at different levels of granularity. Because the user must be able to identify and distinguish automatically-constructed groups, we also address the important sub-problem of producing effective group labels. We conducted an extensive user study comparing PViz to current policy comprehension tools (Facebook's Audience View and Custom Settings page). Our study revealed that PViz was comparable to Audience View for simple tasks, and provided a significant improvement for complex, group-based tasks, despite requiring users to adapt to a new tool. Utilizing feedback from the user study, we further iterated on our design, constructing PViz 2.0, and conducted a follow-up study to evaluate our refinements.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: Miscellaneous

General Terms

Security, Human Factors

Keywords

Privacy, Social Networks, Visualization

Symposium on Usable Privacy and Security (SOUPS) 2012, July 11-13, 2012, Washington, DC, USA.

1. INTRODUCTION

Online social networking systems have existed for many years, but the changing features of these systems, coupled with mass adoption, have exacerbated problems of privacy and presentation management. These changes have created a situation in which boundary regulation [22] is difficult to achieve, and users have difficulty constructing accurate mental models of who the system allows to access what.

In this paper, we focus on the *policy comprehension* problem. Our goal is to assist the user in understanding the visibility of her data in a natural way. Recent work has observed that users' mental models of privacy and publicity in social networks often involve natural subgroups, or communities, within their local networks of friends [9, 13, 23]. The introduction of Facebook's lists and Google+'s "Circles" shows the acceptance of this group-based mental model. However, these groups require manual curation and are underutilized (months after their introduction, less than 5% of Facebook users had created even one list[1]). To address these issues, companies such as Facebook have moved to dynamic "Smart Lists," lists created automatically based on simple queries, like one's hometown, employer or school. Unfortunately, these automatic groupings are noisy and problematic for managing security [13]. Thus, users of these systems require tools for assessing who is in these automatically-created groups and if these groups are acceptable to them, and also for managing the privacy settings applied to these groups. However, existing policy comprehension tools, such as Facebook's Audience View, which allows the user to view her profile as it appears to each of her friends, meet only a subset of these requirements, and are ill-suited for use with this group-based mental model. As this trend toward group-based interfaces and automation continues, and users' online presence continues to grow, the need for usable group-based comprehension tools will become more important.

In this work, we draw a distinction between two common types of tasks that users seek to complete using policy comprehension tools: *single tasks*, in which the user seeks to understand whether a data item is visible to a single friend, and *group tasks*, in which the user seeks to understand whether a data item is visible to a natural subgroup of friends. Given that a user's mental model is often group-based, we anticipate that many tasks performed are group tasks.

[1]http://www.fastcompany.com/1693443/facebooks-big-announcements-dashboards-personal-information-downloads-friend-group-lists

EXAMPLE 1. *Consider Margaret, who is evaluating her privacy settings on a popular social networking site. Margaret would like to keep in touch with John S., a high school boyfriend, and former teammates from her high school cross-country team. In Margret's case a* ***single task*** *would be to determine if her phone number is visible to John. Notice that single tasks are easily resolved using the Audience View tool; Margaret can simply view her profile as it appears to John. In contrast,* ***group tasks*** *prove more challenging. For example, determining whether Margret's phone number is visible to all of her cross-country friends. To answer this question using the Audience View requires Margaret to enumerate every member of the cross-country team, and to view her profile as it appears to each of them.*

In a limited set of cases, rule-based interfaces (e.g., Facebook's "Custom Settings" page) can be used for group tasks. However, this typically requires that the user has explicitly constructed a list containing exactly the members of the group (e.g., "Cross-Country Friends"). In many cases, such as when there is no explicit list, or worse, there are conflicting rules (individual friends or lists assigned to both "Make this visible to" and "Hide this from"), the rule-based interface makes group tasks difficult.

To address the policy comprehension problem, we designed, built, and iterated on a tool, called PViz, which corresponds more directly with users' mental models of privacy. PViz allows the user to understand the visibility of her profile at multiple levels of granularity, and according to automatically extracted natural sub-groupings of friends. To support visual exploration, we also devised a set of techniques to provide concise, human-readable labels for communities within the local network.

We followed an iterative design and evaluation approach. Using our first version of PViz, we conducted an extensive laboratory-based user study comparing PViz to existing policy comprehension tools (Facebook's Audience View and Custom Settings page). Our results indicate that PViz and Audience View achieve comparable results for single tasks. For the more complicated group tasks, PViz provides a significant improvement in user accuracy. Using feedback from the study, we further refined the interface to better suit user needs, incorporating navigational elements and new features to help the user more quickly identify likely policy misconfigurations. Follow-up interviews with participants from the initial study yielded positive qualitative feedback.

2. RELATED WORK

The development of tools to assist average users in specifying, comprehending, and maintaining fine-grained privacy settings is a serious emerging problem in social media. One early study by Acquisti and Gross discovered that while users of social networking sites expressed high levels of concern about their privacy, the same users often did not apply strong privacy policies to their profiles [1, 11]. In many cases, this was due to users' poor understanding of the available privacy tools and the visibility of their profiles. Broadly, the idea of policy comprehension interfaces has been explored in the HCI community, but with less emphasis on social network systems. For example, Nguyen and Mynatt [20] offer the idea of Privacy Mirrors as a framework ubiquitous computing infrastructure.

Recent work has sought to address this problem for social networks. Lipford et al. [16] initially proposed and evaluated the Audience View, which allows a user to view her profile as it appears to an individual friend, or as it appears to a manually-specified sub-group of friends. A variation of this interface, which allows the user to view her profile as it appears to an individual friend (no groups), is currently deployed by Facebook.[2]

Reeder et al. [25] proposed the expandable grid interface for the purpose of understanding and authoring access control policies in file systems, but the interface shares several common features with PViz. The expandable grid allows a system administrator to visualize and modify access control settings using a two-dimensional grid–principals (users) $\times$ resources (e.g., files)–in which any dimension can be consolidated into coarser groups, or *roles*. Recently, Lipford et al. [17] conducted a pilot study comparing an expandable grid interface with an audience view interface in the context of a social network. While the results did not conclusively favor either of the two interfaces, there are other differences. Perhaps most critically, this study assumed a small set of pre-specified friend groups ("Best Friends," "Family," and "Shady Friends"). In contrast, PViz automatically selects and names meaningful sub-groups, based on social circles that are specific to the individual. This work also points out that compact interfaces (e.g., Expandable grids and PViz) are easier to navigate than the more verbose Audience View when there are many audience groups that are of interest.

Indeed, recent work has sought to understand whether there exist groupings of friends that are natural for the purpose of controlling privacy. Lampinen et al. [15] document the phenomenon of group co-presence in online social networking sites. Fang and LeFevre [9] conducted a study in which participants were asked to hand-label their privacy preferences for specific (friend, data item) pairs. They observed that users often expressed homogeneous preferences for friends within the same densely-connected community. Jones and O'Neill [13] conducted a study in which participants were asked to explicitly group their contacts for the purpose of controlling privacy. They also observed that many users considered structural communities when grouping their friends, in addition to other criteria, such as tie strength. Several others have also advocated the use of structural communities for the purpose of controlling privacy [2, 7]. Amershi et al. [3] developed a system for assisting users in creating on-demand custom groups. They found that their interactive machine learning approach worked well when assisting users in creating large, diverse groups. Kairam et al. [14] analyzed group creation and sharing decisions by Google+ users who were both actively sharing content and effectively managing privacy. They observed that these users often created groups based on differing *life facets* and on tie strength. This large body of work supports the group-based mental model on which we base PViz.

Egelman et al. [8] recently proposed a Venn Diagram interface for social network policy comprehension and modification. The interface displays a small number of (possibly overlapping) friend subgroups corresponding to people who have explicitly signed up for a Facebook "network" (e.g., a university or company). The visual display shares some similarities with PViz. However, the approach of defining subgroups based on network membership has some limitations.

[2]This feature is available by clicking the "View As..." button at the top of one's own profile.

The number of available networks is limited and often fails to capture groups of interest (e.g., family). Further, this approach assumes that users will explicitly join all of the pertinent networks, but in practice we have observed that this is often not the case.

Anwar et al. [4] propose, but do not evaluate, another visualization tool for social network privacy settings. The social network is displayed graphically, and mousing over a node in the graph indicates what that person can access in the current user's profile. Rather than presenting a visualization, Liu and Terzi propose computing a single numeric *privacy score*, which communicates to the user the extent to which his privacy settings differ from others' settings [18]. Besmer et al. [5] examine the effects of social navigation cues on users' privacy decisions.

While much work has focused on tools to comprehend and modify privacy settings that already exist, recent work by Fang and LeFevre [9] has also proposed using machine learning techniques to recommend privacy settings based on minimal input from the user.

3. PVIZ OVERVIEW

The PViz policy comprehension tool is centered on a graphical display, which shows the user's social network. Each node in the display represents a semantically meaningful sub-group of the user's friends (a *community*) or an individual friend. Figure 1(a) shows a screen shot of the pilot version of PViz displaying Margaret's social network. Inspecting the display shows that PViz has found five main communities of friends.

To the left of the graphical display, PViz shows a list of profile items for which the user can configure privacy settings. To view privacy settings for a specific item, the user must select the item from the list. In Figure 1(a), the profile item "Other Phone" is selected.

To interpret privacy settings in PViz the user can observe the color of the node (i.e., community) which ranges from 0% visibility (light) to 100% visibility (dark) and is assigned based on the user's privacy selection for a selected profile item. Alternatively, hovering the mouse over a node reveals an explicit numerical pop-up. For example, in Figure 1(a), notice that the node labeled "U. of Alabama" is darker than the node labeled "UGA," indicating that a larger percentage of friends in the "U. of Alabama" community can see Margaret's "Other Phone" than in the "UGA" community.

PViz also includes the ability to view communities and privacy settings at different levels of granularity by zooming in and out. Figure 1(b) shows the process of zooming in on "Brentwood High School," which reveals three constituent sub-communities ("BHS Cross Country," "Photography Club," and "BHS Soccer"). A hierarchical node-link diagram of this type (e.g., [12, 24]) serves the dual purpose of being consistent with both the mental models of "networks" and communities.

In addition to the graphical display, PViz provides several ways of interacting with the social network graph to enhance exploration. For example, the user may search for a friend's name in a search box and the display will automatically center on the node containing that friend. PViz also provides a text box that displays the names of all members of the currently selected node (community).

EXAMPLE 2. *Consider again the single and group tasks from Example 1; both are easily completed using PViz.*

To check whether her phone number is visible to John S. (single task), Margaret first selects the profile item "Other Phone," and then uses the search box to find the node containing John. If this node is either black or white, then Margaret knows immediately whether or not John can see her phone number. Otherwise, she must zoom in on the display. At the individual level (Figure 1(c)), notice that the node representing John is white, indicating that John cannot see Margaret's phone number.

To check whether her phone number is visible to her cross country friends (group task), Margaret starts at the coarsest level of granularity, and selects the profile item "Other Phone." She recognizes that her high school cross country friends are a subset of her high school friends, so she zooms in on the node labeled "Brentwood High School." Zooming in reveals a node labeled "BHS Cross Country" (Figure 1(b)). To ensure that the node contains the appropriate friends, Margaret may select the node, and inspect the list of friends who belong to the community. After locating the "BHS Cross Country" node, Margaret can interpret her privacy settings based on the node's color, or hover the mouse over the node to view the exact percentage.

3.1 Implementation

We have implemented a prototype of PViz in the context of Facebook. After the user logs into Facebook, PViz downloads all necessary data from the user's account. The current user's friend list, neighborhood network graph (the current user's friends and the friend connections between them), and information from the friends' profiles are all obtained via the Facebook third-party development platform.[3] We have also built a screen-scraping tool to download and process the user's privacy settings, which are not generally available via the open development API.[4]

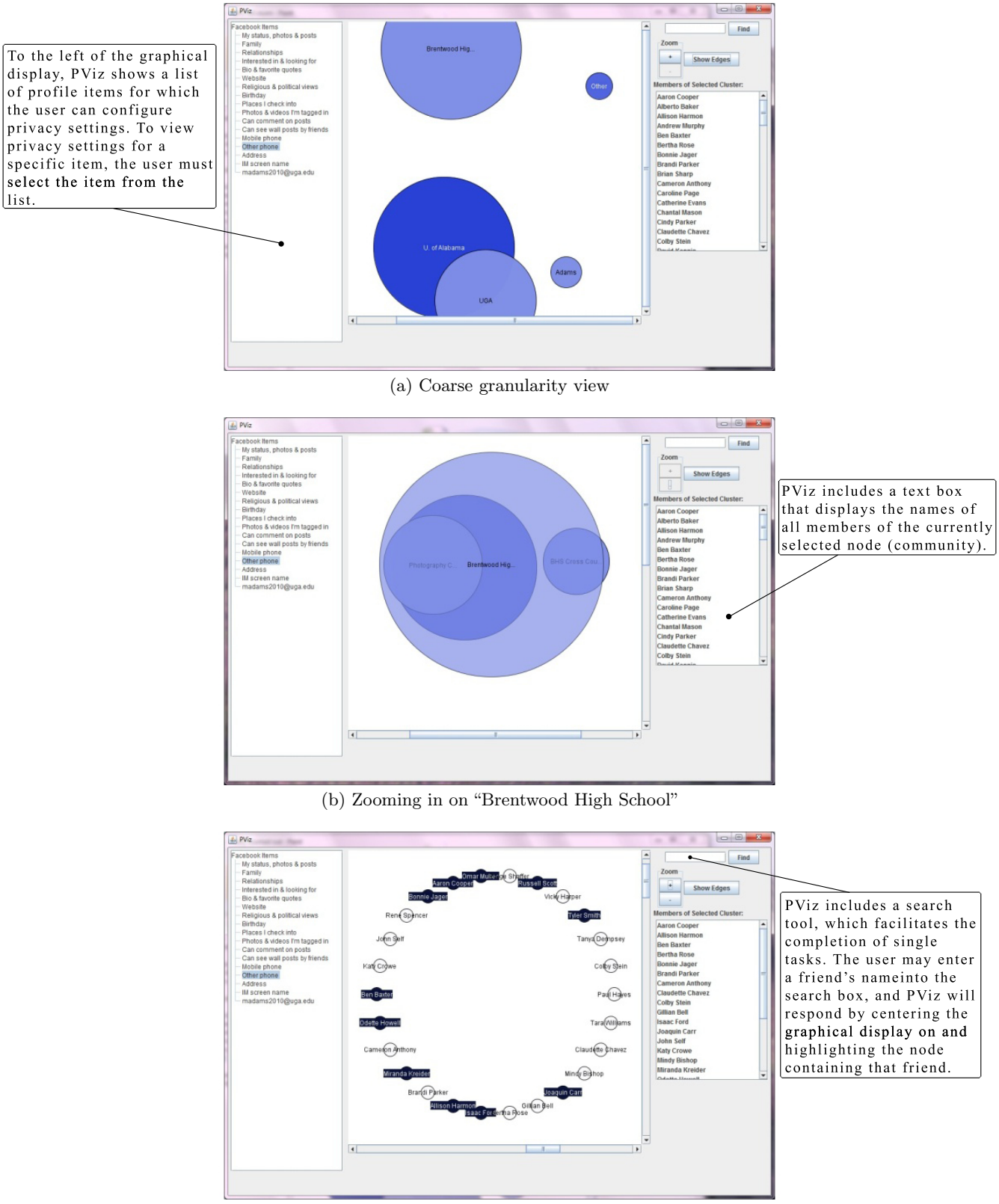
The problem of partitioning social network graphs into communities has been studied extensively [10]. In PViz, our main goal is not to develop new community-finding algorithms. Currently, we apply a common approach based on the idea of *modularity optimization* [19, 21]. When finding communities in a social network, it is often difficult to know the right number of communities ahead of time, and modularity provides a natural parameter-free objective function. In the current implementation, we extract a hierarchy of communities according to a simple recursive process in which (1) the network is partitioned into communities based on maximum modularity and (2) each community is treated as another network that is again partitioned. This is repeated until there is no further partitioning that improves modularity. Of course, the PViz interface is general enough that other community-detection algorithms, as well as explicit groupings provided by the user, can easily be integrated.

After the network is partitioned into communities, PViz positions the nodes on the display using a Fruchterman-Reingold (force-based) layout algorithm[5].

[3] http://developers.facebook.com/

[4] We believe this was in compliance with Facebook's Terms and Conditions, as we downloaded only information about the user's own privacy settings and displayed this information only to that user. In addition, our initial user study operated on synthetic data, which did not require the use of the screen-scraping tool.

[5] As implemented by JUNG, http://jung.sourceforge.net/



(a) Coarse granularity view

(b) Zooming in on "Brentwood High School"

(c) Fine granularity view

Figure 1: PViz allows the user to understand privacy settings at different levels of granularity. Visibility is encoded on a gradient, ranging from from 0% visibility (light) to 100% visibility (dark) and is assigned based on the user's privacy selection for a selected profile item.

3.2 Keyword Labels

The communities in PViz are labeled using informative keywords. The goal of these labels is to enable the user to quickly identify communities of interest. For example, if a user has a group of friends from the University of Alabama, then presenting a group labeled "U. of Alabama" will help her locate this group. While the user may always configure the labels manually, to save time, PViz generates an initial set of labels automatically.

When choosing labels, we assume that each friend has a set of associated *tags* (e.g., the names of cities, schools, and companies), which can be compiled automatically from public profile information.

To support the exploration of the visualization, and to help the user identify the placement of individuals and groups in the visualization space, it is critical to construct informative labels for communities. In designing such a labeling algorithm, we identified two main goals:

1. A community's label should distinguish its members from the rest of the nodes in the graph.

2. Labels should be simple, concise, and easy to understand.

The first goal can be expressed more formally using precision and recall. Let $G = (V, E)$ be a simple unweighted graph with node set V and edge set E. Let $C \subseteq V$ be a community of nodes in G. It is easy to think of a label ℓ as a *query* on the graph, expressed in terms of tags, which returns a subset of nodes $L \subseteq V$. If the query intended to retrieve precisely those nodes in C, then we have $Precision(\ell, C) = \frac{|C \cap L|}{|L|}$ and $Recall(\ell, C) = \frac{|C \cap L|}{|C|}$.

One standard means of combining precision and recall is the F-measure $F(\ell, C) = 2\frac{Precision(\ell,C) * Recall(\ell,C)}{Precision(\ell,C) + Recall(\ell,C)}$.

Definition 1 (F-Measure Labeling). *Given graph G, community C and family of possible labels $\mathcal{L}$, find the label $\ell \in \mathcal{L}$ such that $F(\ell, C)$ is maximized.*

The remaining problem is defining an appropriate *language* for specifying labels in terms of tags. In principle, we could express a label using any logical combination of tags, but this would be complex and difficult for average users to understand. For example, suppose we have three tags: *UGA*, *Tennis*, and *Microsoft*; even if the label $(UGA \vee Tennis) \wedge (\neg Microsoft)$ uniquely characterizes the members of the community, it is not easy to understand. While we describe and evaluate some alternatives below, the pilot version restricts the family of possible labels $\mathcal{L}$ to those comprised of a single tag. In this case, labels are easily selected in time linear in the number of tags.

4. COMPREHENSION EXPERIMENTS

We conducted a set of user experiments comparing PViz to two alternative tools, Facebook's Custom Settings Page (CS) and Facebook's Audience View (AV), which are representative of the state of the art in comprehension tools for fine-grained social network privacy policies.

We recruited 20 participants (9 women) for the study, all students at our university, with a mean age of 23.3 years. Although somewhat restricted, this particular demographic represents a significant fraction of Facebook's user base.[6] In an initial survey, all participants indicated that they had been members of Facebook for at least a year. Self-reported frequency of use ranged from less than once per month to multiple times per day, with most participants indicating that they use Facebook at least once per day. Participants reported a range of experience with Facebook's privacy tools; 70% had previously used the friend list feature, 90% had used the Custom Settings page, and 55% had used the Audience View.

4.1 Standardized Environment

The goal of our study was to compare the utility of PViz to the state of the art policy comprehension tools. An obvious methodology would ask study participants to use each of the three tools to perform single and group tasks related to the visibility of data in their own profiles. Unfortunately, this approach poses several difficult challenges. In particular, in order to evaluate their performance on a comprehensive set of tasks, the participants must have configured their Facebook privacy settings away from the default. According to a recent survey conducted by the Consumer Reports National Research Center, 25% of households with a Facebook account either did not use or were not aware of Facebook's privacy settings.[7]

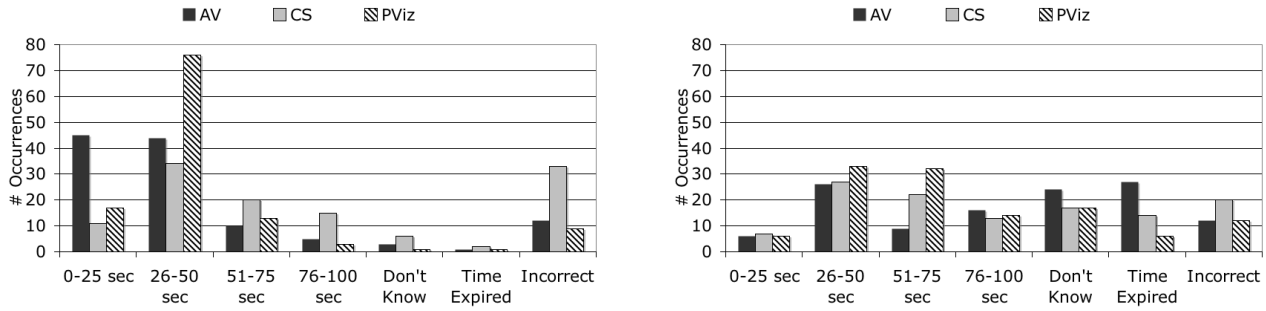
To control for this problem, we instead chose to design an artificial, yet realistic, standardized environment in which to conduct the study. This approach is of course limited because a standardized environment can not perfectly reflect the network of every Facebook user. However, in a subsequent study, described below, participants evaluated privacy settings using PViz and their own Facebook networks.

The standardized environment focused on Margaret, a fictional Facebook user. Her background, social network, friends, profile information, and privacy settings were all created for our study. Margaret had a total of 285 friends (a number consistent with the labeling experiment described below, in which 12 users averaged 297 friends). More importantly, the network was *structurally realistic* as it was based on a real user's network. Margaret kept three Facebook lists of friends: family, graduate school and high school friends. Her privacy settings were configured to allow only a subset of her friends to see each data item. The access control model currently supported by Facebook allows the user to construct both positive ("Make this visible to") and negative ("Hide this from") rules, involving both individual friends and Facebook lists. When configuring Margaret's privacy settings, some data items were given privacy settings that contained *conflicting rules*, meaning that both positive and negative rules were defined for a specific friend or list.

Rather than creating fake Facebook profiles for Margaret and each of her 285 fictional friends, we created local replicas of Facebook's Audience View and Custom Settings pages. We customized them to reflect Margaret's privacy settings and social network by editing the HTML source downloaded from live versions of the two pages. The local pages mimicked interaction with the online Facebook pages almost exactly, although the peripheral functionality was disabled

[6] 30.8% of users are 18-24 years of age as reported by Facebook's Advertising system on February 4, 2011.

[7] http://www.consumerreports.org/cro/magazine-archive/2010/june/electronics-computers/social-insecurity/overview/index.htm



(a) Single Tasks

(b) Group Tasks

Figure 2: Results summary for single and group tasks. The distribution of times for correctly completed tasks is shown on the left of each chart. The right side of each chart displays the distribution of error cases.

(e.g., the ability to click on ads).

When completing tasks, study participants were asked to answer from Margaret's perspective. To realistically model Margaret's interaction with the site, we added several additional cues. For example, a group task might ask whether any of Margaret's high school friends can see her status updates. It is easy to identify one's own high school friends, so to mimic this interaction, we annotated the names of Margaret's friends with numeric flags identifying the groups to which they belonged.

4.2 Tasks

We designed 36 tasks to be completed by every study participant. Specifically, we created two categories of tasks:

- **Single Tasks** Single tasks ask about the visibility of a data item to a specific friend. (E.g., *Can Alice Smith see Margaret's Date of Birth?*) Half of the single tasks required the participant to resolve conflicting rules on the Custom Settings page, and half did not.

- **Group Tasks** Group tasks ask about the visibility of a data item to a group of friends. (E.g., *Can any of Margaret's high school friends see her Status Updates?* or *What proportion of Margaret's friends from UGA can see her Religious and Political Views?*) Using the Custom Settings page, group tasks are easier if the user has created an explicit list for the given group (e.g., *Family*). Half of the group tasks referred to explicit lists, and half did not. For both types of group tasks, we included some yes/no questions, and also some questions that required the participant to enter a percentage.

While we acknowledge that the tasks completed do not cover all possible task types, we believe that single and group tasks are representative of common questions users seek to answer using policy comprehension tools.

For each participant, the tasks were randomly assigned to tools (PViz, AV, and CS). For each tool, each participant was presented with 6 single tasks (3 with conflicts and 3 without) and 6 group tasks (3 with explicit lists and 3 without). The tasks assigned to each tool were then presented in random order. Participants were given a time limit of 1 minute and 40 seconds (100 seconds) per task. Participants had the option of entering an answer for a task, or selecting "I don't know." We measured the amount of time that it took to complete the task, as well as the response accuracy.

Participants completed the study on a desktop computer in a quiet office. Each participant was given detailed background information about Margaret, and presented with each of the three tools in a randomly selected order. For each tool, the study administrator explained the functionality of the tool, and walked the participant through a training task. The study concluded with a post-study survey, soliciting participants' thoughts about the three tools.

4.3 Empirical Results

When evaluating a participant's performance on a task, we used two main criteria: (1) Response correctness, and (2) Total time-to-task (measured in seconds). Figures 2(a) and 2(b) summarize our results for single and group tasks, respectively. For tasks completed correctly and within the time limit ($\leq 100s$), the left-hand side of each chart summarizes the distribution of times. Tasks were considered "incorrect," if the participant (1) selected the "I don't know" response, (2) did not respond within the time limit, or (3) provided an incorrect answer.[8] The right-hand side of each chart summarizes the distribution of error cases.

In analyzing the user study data, we first wanted to determine whether the tool (PViz, AV, or CS) significantly affects correctness. For the purpose of this analysis, we coded any task completed correctly and within the time limit as "correct." We coded all other tasks as "incorrect." To account for any serial correlation within participants (since each participant performed multiple tasks), we ran a logistic regression, clustered by participant. In the regression, the dependent variable was the probability of a correctly-completed task. The results, which are shown in Table 1, show that for group tasks, PViz has a significant positive effect on correctness, relative to AV or CS. (The β coefficients for AV and CS are stated relative to PViz. Since both are negative, this indicates that if we were using PViz, but switched to one of the other tools, we would expect the probability of a correctly-completed task to decrease.) For single tasks, PViz has a significant positive effect on correctness relative to CS, but the difference between PViz and AV is not statistically significant. In all cases, we also considered the order in which tasks were presented (e.g., first, second, etc.) to control for the possibility of learning effects; however, such effects were insignificant.

[8]For percentage questions, we counted a user's response as correct if it was within 5% of the right answer.

(a) Single Tasks

Variable	β	Std. Err.	p
Order	0.0019	0.0173	$p = 0.911$
Tool=AV	-0.4159	0.5037	$p = 0.409$
Tool=CS	-1.6027	0.4260	$p < 0.001$
Constant	2.2623	0.5301	$p < 0.001$

(b) Group Tasks

Variable	β	Std. Err.	p
Order	0.0045	0.0085	$p = 0.597$
Tool=AV	-0.9744	0.2520	$p < 0.001$
Tool=CS	-0.5906	0.2578	$p < 0.05$
Constant	0.7885	0.3178	$p < 0.05$

Table 1: Results of a logistic regression on correctness, clustered by participant.

(a) Single Tasks

Variable	β	Std. Err.	p
Order	-0.3685	0.1382	$p < 0.05$
Tool=AV	-5.5249	2.6497	$p = 0.051$
Tool=CS	12.875	3.8938	$p < 0.01$
Constant	43.583	3.3622	$p < 0.001$

(b) Group Tasks

Variable	β	Std. Err.	p
Order	-0.2711	0.1758	$p = 0.140$
Tool=AV	-1.7204	4.1824	$p = 0.685$
Tool=CS	-1.2489	3.1696	$p = 0.698$
Constant	60.533	4.8585	$p < 0.001$

Table 2: Results of a linear regression on time-to-task, clustered by participant. This analysis only considers tasks completed correctly and within the time limit.

Next, we analyzed the time taken to complete each task. In this analysis, we considered *only* those tasks completed correctly and within the time limit, omitting all others. Table 2 shows the results of a linear regression on time-to-task, again clustered by participant. In this regression, the dependent variable was the time-to-task. For single tasks, we observe that using CS significantly increases the time-to-task, relative to PViz. AV appears to reduce the time-to-task slightly, relative to PViz, but the result is not statistically significant ($p = 0.051$). For single tasks, we also observe a small but statistically significant learning effect. As the value of order increases, time-to-task decreases slightly. For group tasks, neither the tool nor the order has a statistically significant effect on time-to-task.

In the post-survey, we asked participants to assess the tools using three Likert-scale questions. They were asked to respond to each of the following statements on a scale of 1 (strongly disagree) to 5 (strongly agree):

- **Q1:** The tool helped me understand Margaret's privacy settings.
- **Q2:** I enjoyed using the tool.
- **Q3:** I would use the tool on my own Facebook profile.

Figure 3 illustrates the responses to these three questions using boxplots. (The bottom and top of each box indicate the 25th and 75th percentiles, respectively, and the band in the middle indicates the median.) Using a Wilcoxon Signed Rank test (paired by study participant with $p \leq 0.05$), we observed that for question Q1, PViz was rated significantly higher than both AV and CS. For question Q2, PViz was rated significantly higher than both AV and CS. Since the participants were aware of which tool was experimental, it is possible that they inflated scores slightly to please the experimenter. We attempted to control for this by administering the survey anonymously and on paper, as opposed to conducting oral interviews. For Q3, we observed no significant difference between the three tools. We suspect that the responses for Q3 are due to the fact that the PViz prototype supports comprehension tasks, but does not yet support control or policy modification. However, PViz can easily be extended to support policy modification, as we will describe in the conclusion.

4.4 Qualitative Feedback

We received a great deal of qualitative feedback from participants. The most frequent comments were suggestions for improvements to PViz navigation (e.g., zooming using the mouse, a button for zooming all the way out, and tools for moving nodes in the display). We plan to incorporate some of these ideas into the next version of PViz.

In general, the qualitative feedback was consistent with our empirical observations. In particular, several participants drew comparisons between PViz and Audience View for single and group tasks:

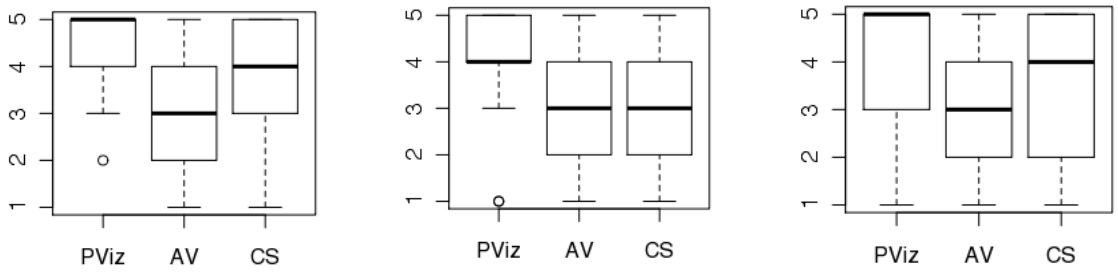
- *Audience view is useful to check a single friend's view of the profile, but hard to see what an entire group has access to. PViz is much more usable and would make me want to set privacy settings rather than remove information entirely.*
- *Pviz was the easiest to use to get a general idea of who could see what information.*
- *It's very difficult to see percentages on Audience view (you have to check each member of a group to get the %) and settings menu.*

One participant also suggested combining features of the Audience View with the automatically-extracted communities of PViz: *Audience view could present the profile as seen by [the] group. If some members of the group can see different fields it's possible to write the percentages of people that can see this field.*

Participants' reactions to the Custom Settings menu were mostly negative, but one participant did indicate that she had developed a strategy involving a limited number of lists: *I have 3 lists (limited, public, family) and put people in groups according to what I want them to see.*

5. COMMUNITY-LABELING EXPERIMENTS

Since informative and meaningful community labels are essential to effectively using PViz, we conducted a small user study to test several community-labeling schemes. We recruited 12 additional participants (again, primarily from our university), and used their actual Facebook friends and networks (59-611 friends, mean = 297). While this is not intended to be a representative user sample, it provides an initial comparison of labeling techniques.



(a) Q1: Helpfulness (b) Q2: Enjoyment (c) Q3: Likely to personally use

Figure 3: Likert-scale responses for user reaction questions

We first downloaded each participant's Facebook neighborhood network, including the graph structure, and for each friend, a set of tags based on the following fields: Current location (City, State, Country), Home location, High School Name, names of companies listed in Work History, names of universities listed in Education History, affiliated organizations, names of Facebook Groups and "Like" pages. We then applied the hierarchical community-detection algorithm described in the Implementation section.

For each community, we showed the participant the list of his / her friends in the community. Then, we asked the question: *On a scale of 1-5, how meaningful is this group (5 = extremely meaningful)?* Finally, we extracted community labels using four alternative techniques:

- **F-Measure:** This algorithm selects the tag that maximizes the F-measure score (see Implementation section).
- **TF-IDF:** This approach is similar in spirit to the F-Measure approach, but is based on an analogy with the standard IR scoring technique. The idea is to count the number of times a particular tag appears in the given cluster (TF), and to normalize by the log of the number of times the tag appears in the entire network (IDF). The algorithm selects the tag with the highest score.
- **Most Common Tag (MCT):** This strawman labeling scheme selects the tag that occurs most frequently among members of the community. Often, labels generated using this approach fail to distinguish members of the community from others in the local network.
- **Logic Rule:** This is another strawman that induces a propositional logic rule, expressed in terms of tags, which distinguishes those friends within the community from those outside of the community. For this experiment, we used the implementation of the RIPPER algorithm [6] as implemented in the Weka package.[9] Although this algorithm uses aggressive pruning, it often produces more verbose labels than the other techniques.

We then displayed the alternative labels to the user, and asked him or her to select the label that best describes the given community, or to indicate "None of the above."

The participants examined a total of 204 clusters, and selected a label for 53% of these clusters. Figure 4 summarizes the results for the cases where the user selected a label. For each labeling scheme, the y-axis shows the proportion of clusters for which it was selected as best, averaged across users. (The error bars show one standard deviation in either direction.) In some cases, two or more of the labeling algorithms produced the same label, in which case it was counted multiple times. As expected, the TFIDF and F-Measure labels were selected more often than the strawman approaches. (Based on a paired t-test, the difference between F-Measure and Logic is statistically significant ($p \leq 0.05$); the difference between F-Measure and MCT is not significant.) Interestingly, when MCT and Logic did produce good labels, those labels were often also produced by the other algorithms. The right-hand side of the chart describes this phenomenon. For example, the average proportion of clusters for which MCT produced the best label, and that label was *not* also produced by F-Measure, was only 0.18.

Intuitively, one would expect it to be easier to generate labels for "good" clusters. To test this intuition, we considered only the clusters assigned a score greater than or equal to the median score awarded by the participant. In this case, participants selected a label for a larger fraction of clusters (64%).

Finally, we were interested in the extent to which we can predict whether a label will be acceptable to the user. This predictive ability would allow us to tell when a generated label is appropriate, and when it may be more appropriate to use a placeholder label that the user may edit later. We considered only clusters for which the F-measure label was selected or the user specified "None of the above," and we tried to learn a model to distinguish the two.[10] We considered a variety of features (precision of the f-measure label, recall, cluster size, cluster depth, and whether the proposed label is also proposed for another cluster), and ran cross-validation experiments, in which one study subject's data was held out for testing during each trial. We observed average predictive accuracies of 70.5% (C4.5 Decision Tree) and 69.2% (Logistic regression).

6. INTERFACE IMPROVEMENTS AND FOLLOW-UP

In response to the feedback we received during the user study, we made several modifications to the interface. These changes can be grouped into two main categories: modifications for convenience in navigating the graphical display, and those made to help the user more quickly understand current privacy settings.

To make navigating the graphical display easier and more convenient, we made several modifications to PViz. Two of our user study participants expressed interest in seeing a list of all communities extracted from their social network,

[9]http://www.cs.waikato.ac.nz/ml/weka/

[10]In this data, 57% of examples have the class label "None."

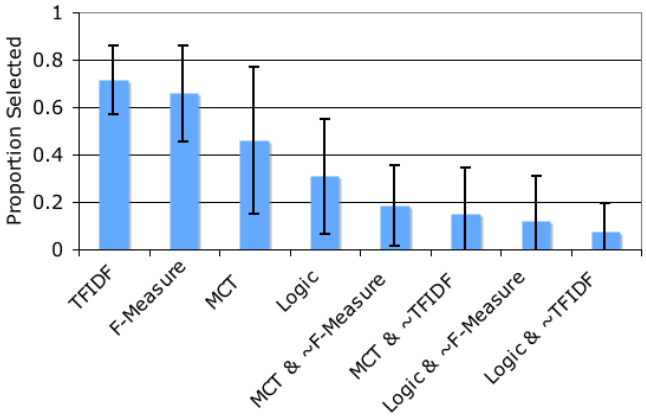


Figure 4: Comparison of Labeling Techniques Based on User Selection

to help them fully understand the hierarchical community structure. To better familiarize the user with this hierarchy, we added a panel to our interface that displays an expandable tree of groups and individuals. Although our tool provides initial labels for communities, we recognize that our labels may not be ideal for each community. Therefore, we allow users to edit initial community names on the expandable tree panel.

When searching for an individual, users reported that they were often interested in viewing individual-level privacy settings for that specific friend, as opposed to simply searching for that friend's location in the coarsest level of granularity. To speed up this common function, we updated the search tool to not only center the display on the friend, but also to take the user straight to the individual level. We believe this modification will improve the time-to-task for single tasks, which, we found to be slightly higher than time-to-task using Audience View, though the difference was not statistically significant. Additionally, at user request, the search box now allows searching on group names. When searching for a group, the display automatically moves to the granularity level that contains the group.

We believe that an effective privacy comprehension tool should call users' attention to possible errors in their privacy configurations. We also believe that errors are more likely to be present in communities that contain outliers. To this end, we modified the color scheme used by community nodes to call attention to nodes in which 10% or less of the members have a differing privacy setting for a chosen item. These groups are highlighted in red, while the homogeneous and evenly-distributed heterogeneous communities remain in the original color scheme.

To further increase salience of outliers in communities that can potentially contain hundreds of individuals, we added a colored background (red and white) to each name listed in the members list, the text box containing members of the currently selected community node. Identifying the outliers in a group with a skewed heterogeneous privacy configuration then simply requires that the user select the node and make a quick scan of the list, to identify friends with dissimilar backgrounds.

To allow users to view their privacy settings in action, we added an "Audience View" display. This display shows the user's Facebook profile as it appears to the selected group of friends or individual friend. For example, when an item is visible to only some members of a group, it is displayed on the page at 50% opacity.

We also made some refinements to the community labeling algorithm. As the difference between the F-Measure and TF-IDF labeling schemes was not statistically significant, we continue to use the F-Measure algorithm in PViz 2.0, but based on the results of our labeling experiment, we set high precision and recall thresholds (0.4 and 0.7, respectively). PViz uses the generated label only if its precision and recall exceed the required thresholds. Otherwise, the system finds up to three representative members of the community,[11] and displays those peoples' names in lieu of a label for the group. Based on informal feedback, we have also reduced the set of tags that are considered to the following: *Last Name*, *Company Name* (from work history), *University Name* (from education history), *High School Name*, *Hometown City*, and *Current Location City*.

We continued to use the modularity-based graph partitioning algorithm, as the labeling experiment suggested that groups produced were often acceptable to the user (mean=3.9, sd=0.4). While we acknowledge that this approach may miss some desired groups, we believe it serves as a reasonable baseline. As in PViz 1.0, other community-detection algorithms could easily be integrated in its place.

Using our improved version of PViz, we conducted a small and informal follow-up study based on users' real networks. We contacted all 20 participants from our initial user study, asking if they would be interested in participating in a short, follow-up study. We received 5 responses, as some participants had left the university and others were not interested in participating. During the follow-up study, each participant was shown their own network and privacy information displayed on PViz and asked to explore their settings and network using the tool.

After using PViz on their own networks, follow-up study participants were asked to respond to several questions about the tool. They were asked to rate the usefulness of each addition individually using a Likert scale of 1 (not useful) to 5 (very useful). The combined mean score of all changes was 4.1 (sd = 0.5), indicating that participants found all updates to be useful.

We also repeated the three Likert-scale questions from the first user study, and added a fourth question concerning the community-finding:

- **Q1:** PViz helped me understand my privacy settings.
- **Q2:** I enjoyed using PViz.
- **Q3:** I would use PViz on my own Facebook profile.
- **Q4:** PViz grouped my friends appropriately.

Participants answered positively on all four questions, with the first three having a mean score of 4.6 (sd = 0.5). The fourth question had a mean score of 4.2 (sd = 0.4).

In addition to the Likert scale responses, we also received qualitative feedback from participants. Most had suggestions for future improvements; one participant wrote: *I'd like to merge and reconfigure some of the groups after they were initially generated.* Another requested that we bring

[11]Currently, we choose the friends in the group with the highest degree in the local network.

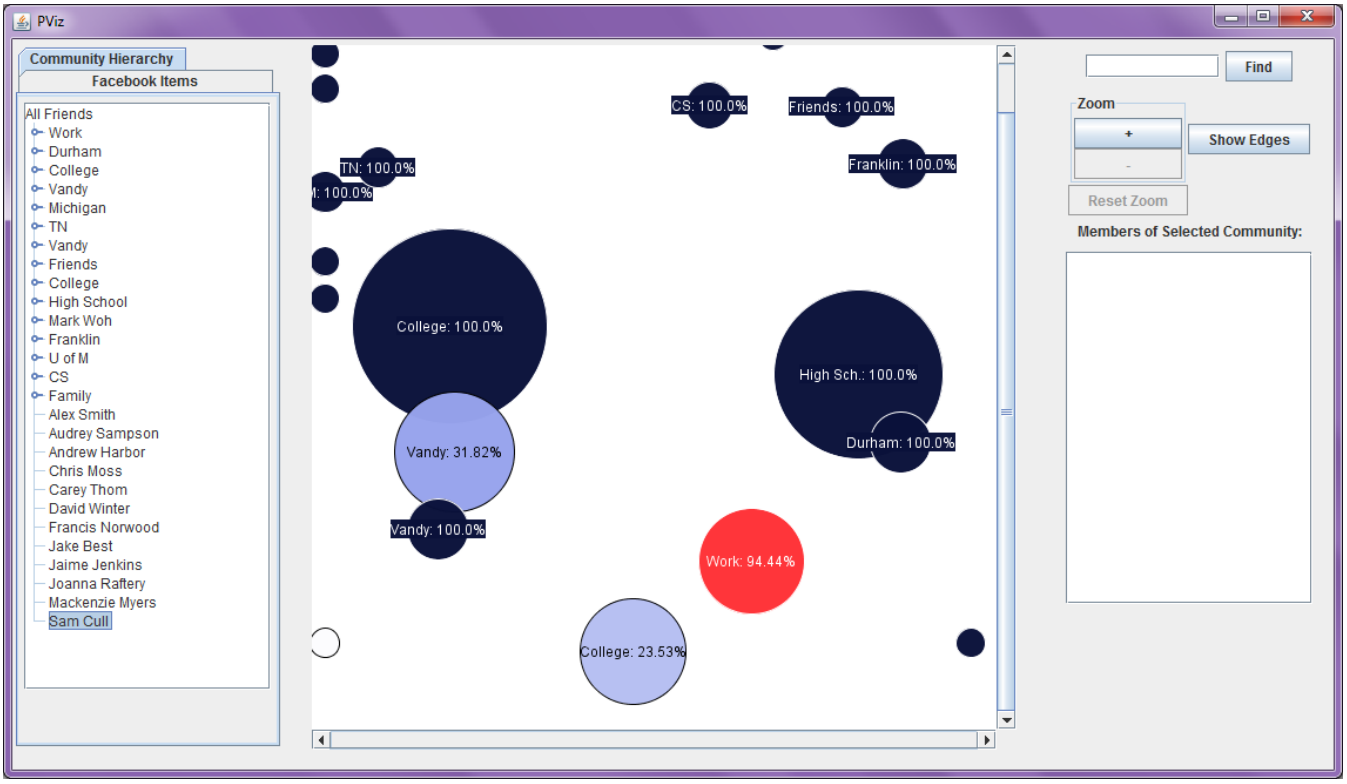


Figure 5: PViz 2.0, with modifications based on user feedback. One community contains an outlier and is colored red.

back the ability to search for individuals without the automatic zoom, and allow the user to pick between the two search modes.

While using PViz, one participant found that he had restricted one individual from seeing several pieces of information, he had forgotten about the settings entirely. Another participant commented: *It reminded me that I don't really manage my privacy settings as well as I should since everyone could access all of my information.*

After using PViz, 3 of the 5 participants indicated that they intended to modify their privacy settings using information gleaned. The other two participants indiciated that they may modify their settings.

7. USER STUDY LIMITATIONS

We conducted an extensive user study comparing PViz to the current state-of-the-art. However, we recognize that the standardized study and small follow-up only evaluate certain facets of the system. Although we believe that this type of evaluation is a pre-requisite for deployment, a broader analysis may provide additional insights, use-cases and limitations.

An ideal evaluation would directly compare the comprehensibility of a user's own privacy settings when viewed using PViz and conventional tools. In such an evaluation, one might ask comprehension questions about the visibility of each subject's own data, both to individuals and groups of friends to which the user was connected.

Although we considered deployment, we found two main prohibitive issues (beyond the practical integration issues with limited APIs and changing Facebook content pages). First, a large number of Facebook users do not configure their privacy settings away from the default. In the default case, all data is visible to all friends, and so the answers to all comprehension questions would be the same. Second, the wide variation of users and networks on Facebook would make creating analogous tasks across users a difficult undertaking.

To combat these issues, we designed a standardized environment in which to conduct our study. The environment consisted of a standardized, fictional user (Margaret), her network and a set of tasks that would enable us to effectively compare performance across tools. This standardized user was designed to be consistent with our user study participants, who were all students at our university. Thus, the standardized user was a graduate student attending school in the United States. User study participants were given a detailed description of her background and her three main social groups. Participants were given time to read this description before completing the study and kept a copy of the description in front of them during the study. Her privacy settings were configured to allow a different subset of her friends to view each item, to ensure that all tasks had unique answers.

The standardized tasks included all relevant and necessary information for completion. For example, a task required the participant to evaluate whether John Self could see Margaret's phone number. In the task, participants were also told of any networks or lists that John was included in, and given any other information that might assist them in locating John in Margaret's social network, such as John's

membership in Margaret's high school cross country team.

We included tasks that asked questions about the visibility of single pieces of Margaret's data to individual friends and to groups of friends. While these types of questions are certainly not the only questions users may seek to answer using a policy comprehension tool, we believe both that PViz supports a variety of alternative tasks, and that the tasks designed represent a large portion of potential tasks.

Because of the wide variance in Facebook users, their friends and networks, the standardized network may have more or less friends or be more or less modular than each individual's own network. However, we feel the network chosen was representative of our user study population, as the network structure mimicked that of a student, and the number of friends aligned well with the average number of friends in participants of our labeling experiment. Additionally, because individual privacy concerns are varied, a potential benefit of using a standardized environment and a fictional user is that they may allow a subject to participate in a more neutral way and to focus on the task itself.

While we have focused on a narrow population for our experiments, generating a corresponding synthetic user, the evaluation we describe could be expanded to other demographics. While we do not have any evidence that the PViz interface would be less useful in other groups, other populations would allow us to validate this belief.

8. DISCUSSION

In this work, we examined limits in policy comprehension tools and identified two main tasks: understanding policy configurations for individuals as well as policy configurations for groups.

While existing tools present solutions to the policy comprehension problem for individuals (single tasks), they fail to scale to groups. As groups are generally the way people model their networks, privacy comprehension tools should support this view.

In aligning a policy comprehension tool with users' mental models, we recognized several requirements: it should allow users to see and identify their groups, validate their settings on groups, and should emphasize potential problem spots in a user's configuration. In PViz, we achieved this by partitioning the user's network into groups and by displaying them graphically with informative keyword labels to allow the user to see and identify groups. To enable the user to validate their settings on groups and to emphasize potential problem areas, we visually encoded these settings through color.

Although PViz is not directly analogous to existing tools, our study revealed that users readily adapted to the tool. It also revealed that PViz was comparable to existing tools for simple tasks, and provided significant improvements for more complex group-based tasks.

We believe the results of both our initial study of PViz 1.0 and the follow-up study of PViz 2.0 motivate the need for policy comprehension tools that provide the user with complete information – information about both group membership and about the privacy policies applied to these groups.

9. CONCLUSION AND FUTURE WORK

In this paper, we introduced the PViz policy comprehension tool for social network privacy. The tool is designed to be more directly aligned with users' mental models of privacy, which often involve natural and user-specific subgroups of friends within their local networks, while allowing users to investigate and assess group membership.

We conducted an extensive user study comparing PViz to the current state of the art. The study indicated that PViz results in significantly better accuracy than existing tools for group tasks and provides support for single tasks that is comparable to the existing Audience View interface. We made further modifications to the interface based on participant feedback, and demonstrated PViz on several non-synthetic networks during a follow-up study.

In designing PViz, we focused primarily on the privacy comprehension problem (resolving one's mental model of privacy and publicity with the existing configuration). There are future opportunities to provide improvements in this regard (e.g., improved community detection and labeling algorithms). However, we believe that PViz also provides a natural platform for privacy control. In the future, we plan to extend the PViz tool to include support for policy modification. We believe that the extension will be straightforward; one possible approach involves attaching drop-down boxes to the various communities in the visual display, as proposed in prior work [8].

As companies' interfaces increasingly support and encourage a group-based mental model and trend towards automation, many of their existing privacy comprehension tools are ill-equipped to handle the added complexity and are becoming inefficient. Unless privacy comprehension tools are reconfigured for group-based settings, these problems will only be exacerbated by users' growing online presence, more potential privacy settings and more automation.

10. REFERENCES

[1] A. Acquisti and R. Gross. Imagined communities: Awareness, information sharing, and privacy on the facebook. In *Privacy Enhancing Technologies Workshop*, 2006.

[2] F. Adu-Oppong, C. Gardiner, A. Kapadia, and P. Tsang. Socialcircles: Tackling privacy in social networks. In *SOUPS*, 2008.

[3] S. Amershi, J. Fogarty, and D. S. Weld. Regroup: Interactive machine learning for on-demand group creation in social networks. In *ACM Conference on Human Factors in Computing Systems (CHI): to appear*, 2012.

[4] M. Anwar, P. Fong, X.-D. Yang, and H. Hamilton. Visualizing privacy implications of access control policies in social networks. In *Workshop on Data Privacy Management*, 2009.

[5] A. Besmer, J. Watson, and H. Lipford. The impact of social navigation on privacy policy configuration. In *SOUPS*, 2010.

[6] W. Cohen. Fast effective rule induction. In *ICML*, 1995.

[7] G. Danezis. Inferring privacy policies for social networking services. In *AISec*, 2009.

[8] S. Egelman, A. Oates, and S. Krishnamurthi. Oops, i did it again: Mitigating repeated access control errors on Facebook. In *CHI*, 2011.

[9] L. Fang and K. LeFevre. Privacy wizards for social networking sites. In *WWW*, 2010.

[10] S. Fortunato. Community detection in graphs. *Physics Reports*, 486, 2010.

[11] R. Gross and A. Acquisti. Information revelation and privacy in online social networks. In *Workshop on Privacy in the Electronic Society*, 2005.

[12] J. Heer and d. boyd. Vizster: Visualizing online social networks. *InfoVis*, 2005.

[13] S. Jones and E. O'Neill. Feasibility of structural network clustering for group-based privacy control in social networks. In *SOUPS*, 2010.

[14] S. Kairam, M. J. Brzozowski, D. Huffaker, and E. H. Chi. Talking in circles: Selective sharing in google+. In *ACM Conference on Human Factors in Computing Systems (CHI): to appear*, 2012.

[15] A. Lampinen, S. Tamminen, and A. Oulasvirta. All my people right here, right now: Management of group co-presence on a social networking site. In *GROUP*, 2009.

[16] H. Lipford, A. Besmer, and J. Watson. Understanding privacy settings in facebook with an audience view. In *Conference on Usability, Psychology, and Security*, 2008.

[17] H. Lipford, J. Watson, M. Whitney, K. Froiland, and R. Reeder. Visual vs. compact: A comparison of privacy policy interfaces. In *CHI*, 2010.

[18] K. Liu and E. Terzi. A framework for computing the privacy scores of users in online social networks. In *ICDM*, 2009.

[19] M. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review*, 69(2), 2004.

[20] D. Nguyen and E. Mynatt. Privacy mirrors: understanding and shaping socio-technical ubiquitous computing systems. Technical report, 2002.

[21] A. Noack. Modularity clustering is force-directed layout. *Physical Review*, 79(2), 2009.

[22] L. Palen and P. Dourish. Unpacking "privacy" for a networked world. In *CHI*, 2003.

[23] S. Patil and J. Lai. Who gets to know what when: configuring privacy permissions in an awareness application. In *CHI*, 2005.

[24] A. Perer and B. Shneiderman. Balancing systematic and flexible exploration of social networks. *IEEE Transactions on Visualization and Computer Graphics*, 12:693–700, 2006.

[25] R. Reeder, L. Bauer, L. Cranor, M. Reiter, K. Bacon, K. How, and H. Strong. Expandable grids for visualizing and authoring computer security policies. In *CHI*, 2008.