Evaluation of Automated Friend Grouping in Online Social Networks

Motahhare Eslami
University of Illinois at Urbana-Champaign
201 N. Goodwin Ave.
Urbana, IL 61801 USA
eslamim2@illinois.edu

Amirhossein Aleyasen
University of Illinois at Urbana-Champaign
201 N. Goodwin Ave.
Urbana, IL 61801 USA
aleyase2@illinois.edu

Roshanak Zilouchian Moghaddam
University of Illinois at Urbana-Champaign
201 N. Goodwin Ave.
Urbana, IL 61801 USA
rzilouc2@illinois.edu

Karrie Karahalios
University of Illinois at Urbana-Champaign
201 N. Goodwin Ave.
Urbana, IL 61801 USA
kkarahal@illinois.edu

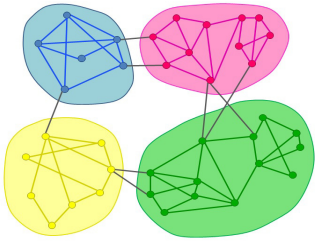
CHI 2014, Apr 26 - May 01 2014, Toronto, ON, Canada
ACM 978-1-4503-2474-8/14/04.
http://dx.doi.org/10.1145/2559206.2581322

Abstract
Managing friendship relationships is challenging due to the growing number of people in online social networks (OSNs). While grouping friends sometimes mitigates this challenge, the burden of manual grouping still prevents OSNs users to create groups widely for privacy control, selective sharing and filtering. In this paper, we present an automated friend grouping tool which utilizes three different clustering algorithms to create groups from Facebook friendship networks. By conducting 18 semi-structured interviews, we investigated the advantages and disadvantages of automated friend grouping in OSNs.

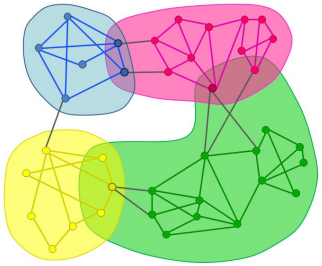
Author Keywords
Automatic Grouping; Clustering Algorithms; Online Social Networks

ACM Classification Keywords
H.5.2. Information interfaces and presentation (e.g., HCI): User Interfaces. H.3.3. Information search and retrieval: Clustering. H.3.5. Information search and retrieval: Online Information Services.

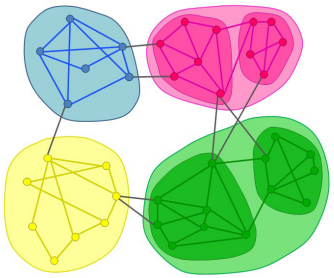
Introduction
With growing numbers of people using Online Social Networks (OSNs), managing and understanding large networks can be a challenge. Grouping friends has been



(a) Disjoint Clustering (Markov Clustering algorithm)



(b) Overlapping Clustering (OSLOM algorithm)



(c) Hierarchical Clustering (Louvain algorithm)

Figure 1. Three clustering methods with different membership attributes used in our grouping tool

suggested as a solution to help OSNs users in controlling privacy, sharing and filtering content. Facebook lists, Google+ circles and Twitter lists are some examples of grouping friends in OSNs. However, previous studies on friend grouping in OSNs [1, 2] illustrated that the high burden of creating groups of friends manually would hinder users adoption. Given the significant burden of manual grouping, these studies suggested automating group creation while allowing users to modify group membership. In this vein, recommendation-based tools such as Facebook smart lists [3], Katango [4], FeedMe [5], SocialFlow [6] and ReGroup [7] have been developed in recent years. These tools suggest recipients for on-demand sharing and filtering of content. The suggestions are based on prior sharing patterns and the content of the intended post.

Such automated recommendation-based techniques can be helpful in social media systems such as email to choose with whom to share a message. However, these techniques discourage users from creating groups for controlling information flow on public and large social networks. These suggestion-based approaches still put a relatively high burden on users to verify friend suggestions one at a time. If one user sends ten messages on an OSN, this requires verifying all of the recipients for all ten messages and may become a hinderance for frequent users of OSNs.

Using automated approaches and allowing for minor user modification to create groups in OSNs is an alternative to the existing recommendation-based grouping techniques. This approach creates fully populated groups from the onset and then allows the user to modify them. One method for creating such groups uses clustering algorithms to automatically detect groups in OSNs. While the feasibility of using clustering algorithms for group creation in OSNs has been investigated before [1], less is known about the benefits and drawbacks of using such automated friend grouping approach within a social media interface.

In this work, we present a grouping tool that automatically creates groups within Facebook using three different algorithmic techniques. This tool creates groups from a Facebook friendship network and then allows for human modification of groups. We conducted a study in which we asked participants to work with our tool and modify their populated friend groups as they wanted. During 18 semi-structured interviews, we investigated the advantages and disadvantages of automated friend grouping in OSNs. They are discussed in the following section.

An Automated Friend Grouping Application

In this project, we chose different clustering algorithms for creating groups automatically. Clustering algorithms can be classified into three categories based on their membership attribute: (i) *disjoint clustering algorithms* where each object can only belong to one group; (ii) *overlapping clustering algorithms* where an object can be a member of more than one group; and (iii) *hierarchical clustering algorithms* which categorize objects in a multi-level structure where one group can be a subset of another group. To have a comprehensive evaluation of automated grouping, we chose one algorithm from each category in our tool. Figure 1 shows a schematic view and the names of these clustering algorithms based on the defined membership attributes.

After implementing the chosen clustering algorithms, we used them to build a Facebook application. We extracted the Facebook friendship network of a user and applied these algorithms on it to create his/her groups of friends. Figure 2 shows a snapshot of our application. The groups created by each algorithm are shown in a separate tab. Each tab is named after the corresponding membership attribute of the clustering algorithm: disjoint, overlapping, and hierarchical.

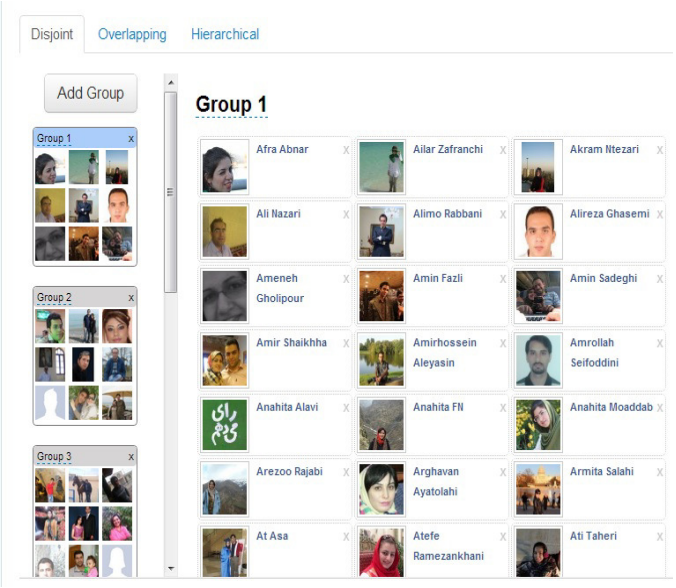


Figure 2. A Snapshot of the Facebook automated friend grouping application

Each tab contains two panels: the *groups panel* (left side) and the *members panel* (right side). The *groups panel* shows the created groups by the corresponding algorithm. By clicking on a group in this panel, the members of that group are shown in the *members panel*. Users can move their friends from one group to another. They can also change the name of a group through both the groups and members panels. At the bottom of the group panel, there is a category named "*ungrouped*" which contains any friends that the algorithm did not place into existing groups. The overlapping and hierarchical tabs offer some additional features. For instance, in the overlapping tab, moving a member from one group to another group would not result in removing the member from the first group. Similarly, in the hierarchical tab, color coding distinguishes groups at different levels of the hierarchy.

In order to evaluate how an automated friend grouping fit into the social media users' intended grouping goals, we conducted a lab study. In this study, we asked people to use our application to investigate the advantages and disadvantages of automated group creation. To date, we have recruited 18 (11 female and 7 male) participants from University of Illinois at Urbana-Champaign. They were from 8 different departments and ranged in age from 18-55. The participants' Facebook friendship networks ranged in size from 139 friends to 1853 friends (μ= 601). All the participants reported using Facebook daily (on average for the past 5.7 years) and the majority of them logged into Facebook several times a day (N=12).

During the study, we asked participants to modify each algorithm's automated groups, specifically for a story sharing task. That is, they should make groups for selective sharing of content. As the first step, we asked them to look over each group and label it based on at least 2/3 of the group members. If a group had no meaning for them, we asked them to delete the group.

Then, that group members automatically went to the ungrouped category. After the first round, participants were asked to come back and review the members of each group individually. During the review process, they were asked to move or delete members when they did not belong to a group, create new groups, or merge the existing groups as necessary. Finally, we asked them to check the members of the ungrouped category to see whether they could find a group for any of them. The participants repeated this process for each tab. To mitigate any learning effects, order effects, and bias toward a specific algorithm, we assigned users algorithms in different orders. Due to time constraints, the participants with large network sizes (n > 500) where asked to work on only one or two tabs. In order to evaluate the automated friend grouping approach, we asked our participants to rate each algorithm based on the groups it had created. We also asked subjects to compare this grouping approach with the current existing grouping techniques in OSNs.

Automated Grouping Interface Evaluation

To assess our automated friend grouping interface, we observed participants' actions during the modification process and encouraged them to discuss any issues or opinions they had about the interface. To have a comparison, we asked them to compare our automated grouping interface with the existing manual or recommendation-based interface in Facebook lists. The overwhelming consensus among our participants was that they preferred this automated grouping interface. For example, one of the participants declared how removing the burden of labelling friends individually made her work easier: *"Suggesting friends by FB is not user friendly as I have to add each person one by one; additionally changing a list of friends is not easy because it needs many clicks! I prefer this user interface that creates groups and then I [can] modify them. It will be faster."*

Similarly, another participant working with the grouping application said: *"if FB had this feature, I would probably use it. When FB came out, it didn't have the list feature and then when it had it, it was hard to do it by hand. So, this version will make it easy to manage my groups of friends."*

Overall, the automated grouping interface was well received. It was especially useful where participants were dealing with a large number of friends. Although existing recommendation-based interfaces suggest friends for each group, users must address each suggested friend one at a time. This method can be helpful in private social media such as email. However, it can discourage the grouping of friends in public OSNs such as Facebook, Google+ and Twitter. Users in our and related studies did not want to dedicate the time and effort necessary for the creation of quality groups.

To further evaluate the interface, we analyzed the application interface for each of the three specific membership approaches. We asked the participants how comfortable they were with the interface on a 5-point Likert scale (1=not very, 5=very). Average ratings are shown in Table 1. We describe the results below.

Clustering Approach	Participants' Rate (1-5)
Disjoint	4.2
Hierarchical	3.8
Overlapping	3.6

Table 1. Evaluation of Automation approaches Interface

Disjoint Interface: Among the three membership interfaces, the disjoint interface was considered the most intuitive by participants. They mentioned the ease of moving friends between groups and traversing groups without difficulty as the main advantages. This

algorithm assumes a friend can exist in only one group. Grouping Sally in either a 'close friend' or 'high school friend' group is easier than placing her in five different groups as might occur with an overlapping algorithm and interface.

Hierarchical Interface: Participants stated that handling the subgroups required extra effort in the hierarchical interface. This added effort made the hierarchical interface less comfortable than the disjoint approach. Designing an interface for hierarchical grouping is inherently more difficult due to the size of OSN friendship networks. More work is needed to manage the groups and subgroups effectively.

Overlapping Interface: While many of our participants expressed a desire for overlapping groups as they worked on the disjoint approach, the overlapping interface introduced an extra layer of complexity. One of the participants stated: *"I like the idea of overlapping as it's useful but it gets confusing when the number of people grows and makes it hard to put one person in different groups.''* Similarly, another participant said: *"it [the interface] makes it complicated to put people in multiple groups."* One of the participants offered graphical suggestions to manage multiple group membership: *"it would be better if you made the overlapping feature in the visualization better so that I [can] understand who is in which groups.''* We plan to iterate on our design to better manage multiple memberships across groups through visualization.

Automation Effect

Previous studies on friend grouping [2] showed that different techniques for grouping friends manually affect the final groups one person creates. Although in this study we used the same technique for modified groups in each algorithm, different original algorithmically created groups might affect the final groups a user curates. To examine whether such an effect exists, we compared the final groupings a user created across the different algorithms. As the groups generated by hierarchical clustering in the lowest level were disjoint, we were able to compare those groups with the ones created by the disjoint algorithm after the modification process. Using the BCubed metric [8] as a comprehensive metric for comparing two different groupings, we found on average a 14% difference between two groupings a user creates from the same friendship network. This relatively high difference shows that the automated friend grouping algorithm influences the final groups the user creates significantly. Therefore, choosing an appropriate automated friend grouping technique is very important in designing a grouping tool for OSNs.

In order to understand the possible causes of the automation effect, we compared the associated groups in two different groupings and explored the ones with the highest difference. We found the following factors resulted in automation effect: (1) *following what algorithms create*: during the study, some participants declared that they kept some groups as the algorithm created them. They also said that if they had started to curate groups manually they might not create those algorithmically created groups by themselves. However, they stated that they liked those groups and kept them; (2) *hierarchical structures in social relationships*: one other cause of the automation effect is the difference between the hierarchy levels of the groups created by different algorithms. For example, when one algorithm generated a group for family, the other one

created two groups for it. Since the participants in many cases did not change the groups significantly, the final groups for the same friends would be different. For example, a participant named one group 'family' in one algorithm while he labelled the same friends as two groups of 'my family' and 'my spouse's family' in another algorithm; and (3) *user's uncertainty*: users' doubt about the intimacy level of some friends and the uncertainty in organizing some friends resulted in ambiguity for some participants. Due to this uncertainty, they categorized some friends differently when modifying different algorithms.

Conclusion and Future Work

Given the significant burden of manual and recommendation-based grouping in OSNs, we presented an automated friend grouping tool which utilized three different clustering algorithms to create groups in a Facebook friendship network. Conducting a 18-users lab study and asking participants to compare our automated friend grouping interface to the existing manual or recommendation-based grouping interface in Facebook lists, we found an overwhelming consensus between our participants in preferring this tool. However, we found out a relatively significant automation effect in group creation which should be considered as an important factor when designing automated friend grouping applications.

We plan to extend this project by analyzing the data we have collected about the groups modified by the participants. Exploring the features of human-curated groups such as size and structural attributes is one of our goals. Since having the real labels of groups in online egocentric social networks without asking people is almost impossible, in this study we gathered valuable data which we want to use for evaluating the current group scoring metrics that are used in assessing clustering algorithms. We believe this project is a step toward understanding the advantages and disadvantages of automated friend grouping in OSNs.

References

[1] Jones, S. and O'Neill E. Feasibility of structural network clustering for group-based privacy control in social networks. In *Proc. of SOUPS 2010*, ACM press (2010), 1-13.

[2] Kelley, P. G., Brewer, R., Mayer, Y., Cranor, L. F., and Sadeh, N. An investigation into Facebook friend grouping. In *Proc. of INTERACT 2011*, Springer (2011), 216–233.

[3] Ross, B. Improved Friend Lists, Sep 2011.

https://blog.facebook.com/blog.php?post=10150278932602131.

[4] Kincaid, J. Kleiner-Backed Katango Organizes Your Facebook Friends Into Groups For You, July 2011.

http://techcrunch.com/2011/07/11/kleiner-backed-katango-automatically-organizes-your-facebook-friends-into-groups-for-you.

[5] Bernstein, M. S., Marcus, A., Karger, D. R. and Miller, R. C. Enhancing directed content sharing on the web. In *Proc. of CHI 2010*, ACM Press (2010), 971–980.

[6] MacLean, D., Hangal, S., The, S. K., Lam, M. S. and Heer, J. Groups without tears: mining social topologies from email. In *Proc. of IUI 2011*, ACM Press (2011), 83–92.

[7] Amershi, S. , Fogarty, J. and Weld, D. S. Regroup: Interactive machine learning for on-demand group creation in social networks. In *Proc. of CHI 2012*, ACM Press (2012).

[8] Amigó, E., Gonzalo, J., Artiles, J. and Verdejo, F.. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Journal of Information Retrieval* , 12(4):461–486, 2009.