Automated Community Detection on Social Networks: Useful ? Efficient ? Asking the users

Remy Cazabet
IRIT
Toulouse University
Toulouse, France
cazabet@irit.fr

Maud Leguistin
LISST-CERS
Toulouse 2 - Le Mirail
University
Toulouse, France
mleguistin@gmail.com

Frederic Amblard
IRIT-UT1
University of Social Science
Toulouse, France
frederic.amblard@univ-tlse1.fr

ABSTRACT

In most online social networks, with the increasing number of users and content, the problem of contact filtering becomes more and more present. The recent introduction of such features in online social networks – for instance, Circles in Google+ or Facebook Smart lists – shows that it is a problem they are confronted to. In this paper, we explore this question through multidisciplinary aspects. First, we discuss about this issue of groups management in the context of social networks. Then, we present several techniques from the state of the art to automatically find meaningful groups of contacts in a user's contact list. Finally, we asked Facebook users to evaluate these solutions on their own Facebook network, both to compare the solutions among themselves and to assess how pertinent the best ones are according to them. The conclusions of this study is that a network analysis approach can strongly improve the efficiency of an automated detection of groups on networks, which could be used, combined with profile data extraction, to design intelligent management of groups of contacts.

Categories and Subject Descriptors

H.3 [**Information Systems**]: Information storage and retrieval; J.4 [**Computer Applications**]: Social and behavioral sciences

1. INTRODUCTION

Online social networks are becoming everyday more and more important, growing both by their number of users and by the importance they take in users' life. As they become part of our societies, they also become an interesting research topic for scientists from as different fields as social sciences, complex networks analysis and data mining.

One feature shared by most online social networks is the possibility to explicitly define a list of contacts to whom the user is more closely related than to the average user. This list of person can have different names or meanings, and can correspond to different implementations:

- On Facebook, it is a list of "friends", which implies a mutual acceptation.
- On Twitter, there are more precisely two lists, one of followers and one of followees, which do not imply mutual acceptation. It can happen frequently on Twitter to follow a complete stranger, while usually the list of friends on Facebook is more restrictive.
- On Google+, these two notions exist together.

As more and more people use social networks for more and more activities, new needs appear, as well as new issues. One of these concerns the aggregation of an user's multiple communities over a single network. The literature in Sociology, in particular Goffman [10], underlines that social actors do not interact similarly with all their contacts. One do not communicate in the same way with his family, his co-workers and his friends. Similarly, social networks users do not necessarily want to share the same information with their close friends and with some half forgotten summer contact.

Facebook was probably the social network the most affected by this problem. The phenomenon of friends grudgingly accepted is well known, and when interrogated, users frequently report [11] that they are not free anymore to say what they want on Facebook, because of, for example, their parents, or their work superior being friend with them on Facebook. When Google+ –the direct competitor of Facebook– was launched, one of the main new feature introduced was the possibility to create and manage "circles" of friends. One can easily decide to create and edit circles corresponding to different communities or targeted audience, and then assign each of his contact to one or several "circle". Actually, Facebook also proposes a quite similar possibility, but less known and less used.

One idea of this paper was to interview Facebook users to know their position about this problem on the most largely used social network. We wanted to ask to users how they deal with this issue in their everyday usage, and how much it matters to them. The first part presents the results of these interviews.

The second part of the paper consists in proposing a tool to help users to manage their groups of friends, by automatically detecting them from the user' contact list. We proposed several way to do so, and ask users to give their feedback on these solutions. The final part of the paper presents these results.

WI&C'12 April 16, 2012 Lyon, France
Copyright 2012 ACM 978-1-4503-1189-2/12/04 ...$10.00.

2. FACEBOOK, USERS AND THEIR CONTACTS

In the first part of the survey, we asked Facebook users several questions concerning their usage of Facebook, and more specifically on the means they use to deal with this problem of having several types of contacts gathered on a same platform, without the usual natural separation between them.

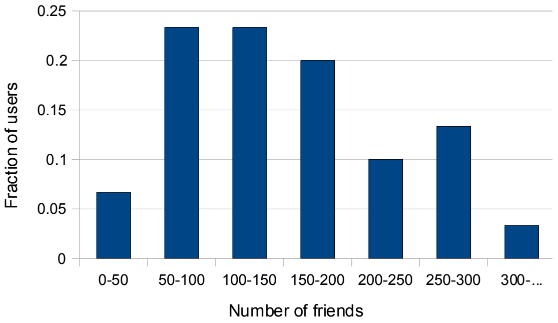

Figure 1: Repartition of users by number of contacts

2.1 Panel of users

We got a panel of 30 users, chosen among people of different age and occupation. The panel is not exactly representative of Facebook users, but represents a certain diversity. 50% of users are men and 50% women. 60% of users are between 20 and 30 years old, 10% under 20 years old and 30% above 30. On figure 1 we give the repartition of users in term of number of friends. 24% of users had already defined manually lists of friends in Facebook. 30 users is not a very large panel, but the process of comparing and evaluating propositions of communities is a rather long one and, furthermore, this number is sufficient for clear tendencies to appear.

2.2 Sociological analysis of users' habits

Many works have been done in sociology recently on social networks, sometime related to this topic [1] [6]. Dominique Cardon [3] for example realized a cartography of web 2.0 websites, using criteria such as "fiction oriented" or "subject oriented". According to Cardon, fiction oriented platforms are platforms on which people are, for example, anonymous, or platforms for fictional activities such as video games. On these networks, controlling precisely one's communication is less important, as we are free to create a fake personality, and real life will not be directly impacted by our online actions. On the contrary, Facebook is classified as a subject oriented one. Users want to be able to communicate with people they know. In order to exist on these networks, to be credible, users need to share, to exchange. More precisely, Facebook –similarly to dating sites– is based on the idea of real but chosen personality. One needs not to lie on these websites, as the truth is likely to be known sooner or later. As exemplified by [4], there is a loop between the real and the virtual life. The computer usage is inseparable from social practices. But if people cannot lie, they can select what they decide to publish, and therefore can choose to show themselves as they want to be seen. Here appears the notion of users' groups of friends: one would want not only to control what they share, but also with whom.

2.2.1 *Contact list management*

We first asked to the interviewees how they use to manage their contact list on Facebook. Two different behaviors are observed: some users tend to accept everyone or nearly everyone as a contact, and then "clean up" their list regularly. Another behavior consists in refusing to add some people, when they are not considered as close enough. Of course this latter behavior can be more problematic with people that the user effectively knows, as it can be seen as a direct refusal. The fist method is socially more acceptable, as the contact is not directly informed of the removal, and might even never notice it. Our first observation is therefore that the majority of our subjects use one or the other of these two methods. It proves that users are actually aware of this problem of management of their public information. For most users, contacts on Facebook must only be people they already know. They are not virtual relations, but existing ones with whom Facebook is another communication media. However, they could be close relations, as well as more distant ones.

2.2.2 *Communication by group of friends*

We asked our subjects if, first, they knew that Facebook was proposing a tool to create groups of friends, and then, if they were using it, and how much they thought such a tool could be useful to them. Approximately two thirds of the subjects knew about this possibility, and a majority rated this type of tool as interesting or useful. However, very few users declared to use these tools on Facebook, around 20% of them, and most of them declared that they nearly never use their lists to restrict their publication targets. It is not because they do not care, but they found simpler ways to handle it, for example by using different communication services for different usages. Most people use status update only in the case of information they consider as public, and otherwise, use the private messages or instant messaging service. Of course, this implies sometimes to refrain from posting status updates if the user do not want it to be seen by some users. An interesting remark is that young users (18-24), even if they have in average more Facebook contacts than older ones, declare themselves as more concerned by the problem of selected diffusion of information. They have more expertise in using the platform, and therefore use more the group management tools.

2.2.3 *Conclusion*

As it was studied in [11], after a socialization or initiation period, the public communication and the management of the relations become new competencies in the usage of the platform. By a process of trials and errors (with, sometimes, important repercussions on users' life), social networks' users learn to identify and construct a hierarchy of the different communities composing their relations. According to their degree of expertise in the platform's usage, they restrain their communication, use different communication media or use the tools provided to efficiently manage their communication. Social networks, and Facebook in particular, are tools used to communicate with close friends but also to maintain active links that would have been lost otherwise. There is a continuity, not an opposition, between real and virtual, and, as a consequence, far from generally accepted ideas, most users are vigilant on the question of their communication.

3. A TOOL TO AUTOMATICALLY DETECT COMMUNITIES OF FRIENDS

We have seen in the previous section that in many cases, it can be helpful for social networks' users to publish messages for a restricted audience. Often, the targeted group is composed of, either a group of close friends (compared to more distant ones) or a group of contacts sharing a common background. The idea of the following is to automatically identify some groups, in order to help the users to create and manage them. Users tend to have large contact list, and proposing relevant communities in them, or automatically updating them, can help the users, while doing the same thing manually is a tedious task. Finding close friends can be achieved by counting the amount of communication between persons on the social network. In cases where this information is not relevant, we do not really have other means to find it out. On the contrary, for the other type of groups, we can have more information, that we will use hereafter to try to detect automatically these groups.

3.1 Existing tools

A few Facebook applications already exist to identify automatically groups of friends in a user's contact list, namely Friend Sets, Friend Wheel and TouchGraph. However, no information is given about the algorithms used by these applications, but they seem not to use current state of the art algorithms and, notably, do not allow overlap of friends: each friend must belong to one and only one community. There is also one scientific work on the topic, the "fellows" application [8]. However this work is using only a custom community detection algorithm which is not compared to other ones from the state of the art. This application do not provide any visualization of the communities, but only lists of names, which can make the evaluation quite hard with large networks of friends.

3.2 Proposition of solution

In the field of complex networks analysis, there is a problem which is known for a long time, but which attracted very recently a lot of interest, namely the automatic detection of "communities", or groups, in networks. In this context, there is no hypothesis made about the nature of the network. What is studied is the topology of a network, its nodes and edges, and only them. The goal is then to define an algorithm, which will be able to find in this network several groups of nodes closely knit together, and more weakly related to the remaining of the network. It must be observed that there is no universal definition of what is or is not a good community. The difficulty of the problem and the large number of applications conducted to a very large amount of proposed algorithms. These algorithms use very different techniques, as for example the modularity [9][2], clique aggregation [12], Network Compression [13] and many others. Some comparisons have been done between these algorithms, the most frequently on artificially generated networks, showing that some of them were more efficient than others, but there is not (and there will probably never be) an algorithm that is the best for all kind of networks.

In this paper, we propose to use some of these algorithms to try to find relevant groups of friends among the contact list of social networks' users. We decided to work on Facebook, as it is one of the most widely used social network, furthermore proposing a complete API allowing us to obtain many information about the user's friends network. By using the rich information from users' profiles, we will also be able to try to identify the meaning of the obtained group (for example: family, co-workers,....).

3.3 The network of friends

In a first step, we want to represent the contacts of the user as a network. To do so, we simply query Facebook to retrieve the user's list of friends. Each friend will then correspond to a node in the network. The user himself does not appear in it. In a second step, we will query Facebook to obtain all the friendship relations between these nodes. Each of these relations will be represented as an edge in the network. For example, if the user A has 3 friends, B,C and D, and that B and C are also declared as friends on Facebook, the network retrieved for user A will contains 3 nodes (B,C,D), with an edge between B and C. It is a classic ego-centered network at one hop. Note that A does not belong to the network. Otherwise, he would be connected to all nodes in the network, which would not add any meaning to it and might disrupt the algorithms.

3.4 Finding communities

In a second step, we run some community detection algorithms on this network. These networks are not directed, not weighted, and usually of relatively small size. Therefore, nearly all the classic community detection algorithms can run on them, without any specific operation. However, we were interested in one property of some algorithms: the ability to detect overlapping communities. It is well known that in social networks, when we have to define communities among a person's friends, some of them will belong to several communities. For example, one person from your family can also be part of your high-school friends, or one of your co-workers might also be part of a non-related group of friends. It can therefore be helpful to use algorithms able to deal with this overlap issue. However, most proposed methods do not allow this possibility, and they have been less thoroughly studied. We therefore decided to compare methods with and without this overlap possibility. We know that methods with overlap can potentially give better results than methods without, however, as the later ones have been more studied, the best methods without overlap could still give the best results. The idea was to propose to Facebook users the solutions found out by the algorithms, and to ask them to rate their efficiency. The chosen algorithms were:

3.4.1 InfoMap

This method [13] based on network compression, has been chosen because it is frequently considered as the most efficient method without overlap. It has been widely tested, and usually outperforms other algorithms both on generated networks [7] and on real world networks.

3.4.2 CFinder

Cfinder [12] is the most widely known method to detect communities with overlap. This method is quite simple, using aggregation of cliques to form meaningful communities.

One particularity of CFinder is that it needs a parameter, k, which represents the size of cliques the algorithm will search for as its atomics components. Here, we used a value of k of 4, because CFinder is known to be quite inefficient with a value of 3 (discovered communities are in this case too large), and here, 5 is clearly too high (all nodes not belonging to a clique of 5 nodes would be ignored)

3.4.3 iLCD

This method [5] proposed recently, is based on multi-agent systems. Each community is represented as an agent trying to expend on the network by integrating nodes, while trying to remain distinct from other community agents. This method has some interesting properties in our case: first, it is one of the few methods, with CFinder, able to detect overlapping communities. iLCD is also a method which was designed to be efficient on social networks, and can run on dynamic networks: we could use it, when new Facebook friends are added by a user, to integrate these new friends in the existing communities without computing everything from the beginning again, which can result in having a result strongly different from the one at the previous step.

3.4.4 Connected components

We can consider that the most basic form of community detection is to consider each connected component of the network as a community. We will use this method as a comparator, a null hypothesis, to see if other methods really improve this result, or not.

3.4.5 Facebook lists of friends

Facebook introduced a few months ago (2011) a new feature called smart lists. The aim is to try to create lists of related friends according to users' profiles. Therefore, it will automatically create a community including all of your Facebook friends who went to the same high school as you, and name it with the name of your high school. It will try to do the same with your family, your co-workers, and a few other categories. Even if the mechanism has not been officially given, it seems that no network analysis is used, and that information are only based on crosschecking of profile information. In the following, we will consider each list as a community. The problem is that some people have also defined their own lists of friends, or modified manually these smart lists, and we cannot know, by using the API, if the lists were automatically generated or user-created. Therefore, we will ask to subjects if they use the Facebook lists, and consider the results accordingly.

3.5 Naming communities

Many people fill their profile with information about their past and present occupation, like the high school they attended, their colleges, current and past employers, and so on and so forth. By crosschecking these information according to the persons grouped together, we can try to guess what a group correspond to. To do so, we extract from the profile all the information that could be meaningful, namely: high-school information, college information, past and current employers, family name, city of birth, current city. For each term and each community, we count the number of persons whose profile include this term. If the term appears more than once, we consider it as potentially meaningful for this community. We then compute a relevance score, as detailed hereafter:

We first compute $tfc_{t,c}$, the term frequency in community for term t in community C

$$tfc_{t,c} = \frac{n_{t,c}}{|C|}$$

Where $n_{t,c}$ represents the number of persons of C having t at least one time in their profile, and $|C|$ the number of persons in community C.

Then, we compute tfN_t, the term frequency in the whole network as

$$tfN_t = \frac{n_{t,N}}{|N|}$$

Where N is the set of all persons in the network.

The relevance score of a term t for a community C is now simply defined as

$$relevance_{t,c} = \frac{tfc_{t,c}}{tfN_t}$$

This formula represents that a term appearing frequently in the profiles of people grouped in the same community is probably representative of it. And if this term appears mostly in this community, it is more representative than if it appears frequently for profiles outside the community.

In order to limit ourselves to the most significant terms, we keep for each community the 3 terms having the higher relevance as representative terms. However, as we ask for a term to appear at least twice in a community to keep it, it is possible to retain no representative term for a community.

3.6 Visualisation

As we wanted users to evaluate the solutions proposed by the different algorithms, we had to develop a graphical interface to help users to visualize these communities and their representative terms. We have chosen to create a Facebook Application, fully integrated to Facebook to simplify the testing process. One only needs to connect to Facebook and launch the application to automatically see the solutions and explore them visually. Figure 2 presents the 2 available views. Each large circle represents a community. The small colored circles inside them are the user's friends. On the right pictures, we see representative terms appear in the middle of each community. One of the problems was to represent the overlap between communities. After several trials, we decided to draw each friend only once, even if he belongs to several communities, in order to keep a coherent display. The node is placed in the most relevant community according to its number of edges with each community it belongs to. His overlap is then represented as a splitting in color: a unique color is attributed to each community, and the node belonging to several communities will be colored accordingly. Then, an edge is used to connect two communities having common nodes. This visualization could of course be improved, but has proven clear enough to be used by most people.

4. TEST PROCEDURE

We asked to the same panel of users than before to try the different algorithms, and, through different type of questions, we asked them to evaluate the algorithms, both on a global level and on more precise aspects. Users were not provided any information about the tested algorithms, anonymized name were used.

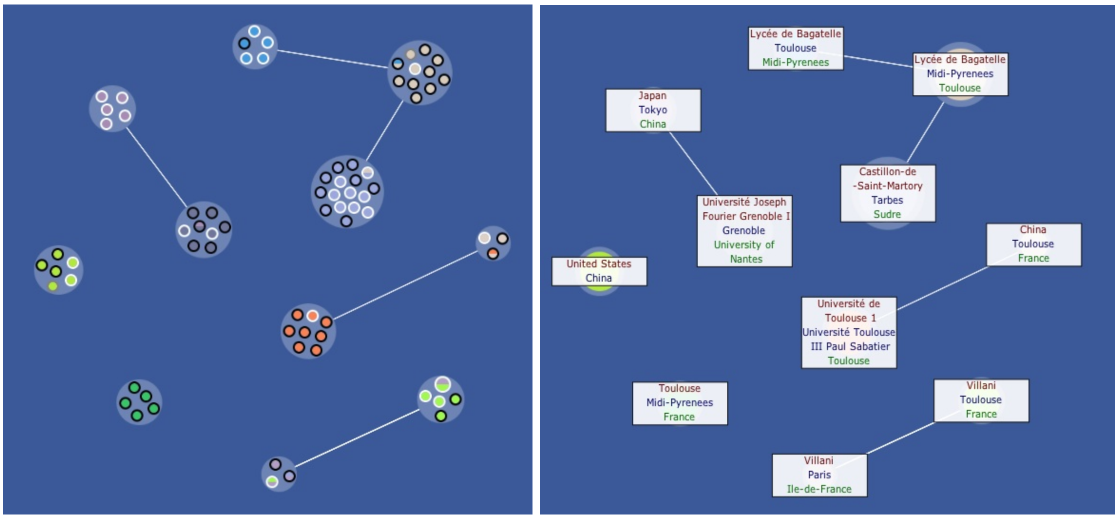

Figure 2: Interface of the Facebook application. Small circles represent the user's friends, larger circles his communities. The top one is without the community names (to explore whose contact is in which community, further informations are displayed interactively)

4.1 Ranking and scoring

First, users were asked to test every algorithm on their network, to compare them, and then to give them:

- A rank (1 being the best algorithm). This rank was unique, in order to force users to choose between algorithms.
- A score, for which we explicitly described the meaning, as follows:
  - 1: The solution is incoherent and/or incomprehensible
  - 2: The solution is bad, but there are some good things
  - 3: Average Solution
  - 4: The solution is good and logical, but there are a few mistakes
  - 5: The solution is perfect

The aim of the score is to evaluate if there is really a strong difference between algorithms, as algorithms ranked 1 and 2 might be quite similar for example, while there might be a strong difference in term of quality between 2 and 3, for instance.

4.1.1 Average results

On picture 3, we can observe the average rank and the average score for the different solutions. The two charts seem to lead to the same conclusions.

- The connected components algorithm is, obviously, the worst algorithm. With an average mark of 2.2, it's considered by most users as bad. Results are not incoherent, but in most cases, most of the user's friends are aggregated in one same community. Therefore this algorithm does not provide enough discrimination among communities.
- The result given by the Facebook list of friends is not better in average. Actually, we must distinguish the results given by persons who had defined lists on Facebook from people who have not, as we cannot distinguish between them. Among the 7 people who have defined their own lists, the average mark was 3.43. Among the others, the average mark is 1.78. As expected, there's a large difference between the two. This means that for people who only have the lists automatically generated by Facebook, they consider this solution as even worst than a simple analysis of the connected components. It was marked as incoherent (mark 1) by 40% of them. We can probably explain these poor results by the way they are obtained, namely by only using profiles cross-checking. On the one hand, many people do not fill their profile completely, or do not keep it up to date. Therefore, these people are obviously missing in the community they should belong to. On the other hand, some people went to the same place –in particular, same high school or college– but are not linked for these reasons. For example, for a person who have not moved a lot since his childhood, the community corresponding to his high-school will frequently contains several of his co-workers, persons of his sport club, and other people of different ages that he could not have met in high-school. Finally, this solution is unable to find more informal groups, which are not specifically related to a place or an activity.
- iLCD and Infomap have very similar average scores and rankings. The average score is around 3.5 for both of them, which is clearly better than the other solutions. However, there is some disparity among their results, which we will detail later.
- CFinder results are more or less in the middle between the results of the best and the worst algorithms results. This is interesting, because it shows not only that the community detection algorithms work well compared

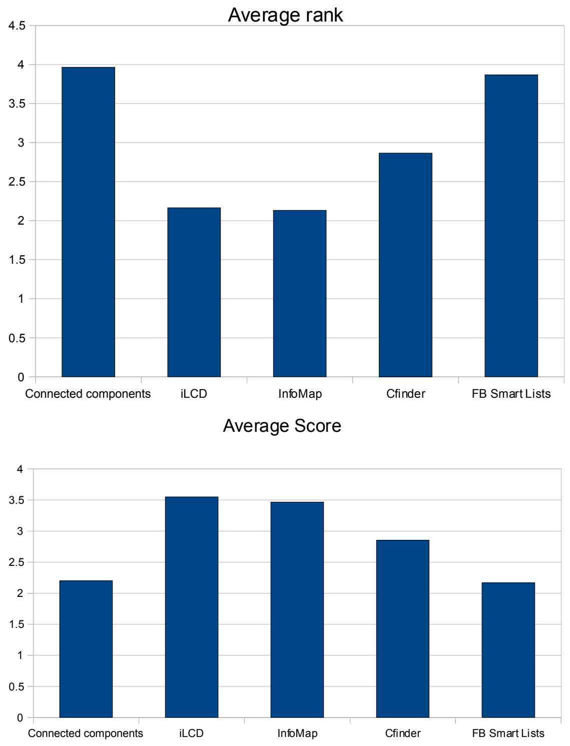

Figure 3: Average rank (lower is better) and score (higher is better) for the tested solutions

> to more simple results (lists, connected components), but also that some community detection algorithms are clearly more efficient than others.

The interesting point here is also that an algorithm which was not able to deal with overlap of persons (InfoMap) gave better results than an algorithm that can. Although this problem is an important one on social networks, it is more important for users to have a globally relevant splitting, with a few errors due to overlap problems, than a solution with overlap but missing some important distinctions between groups.

4.1.2 *Insights into the results*

We were first interested in studying the repartition of scores for each algorithm. When we see an average mark of 3, we do not know if it is due to a lot of medium marks, or an equal number of good and bad marks. In figure 4, we can see the differences. Connected components has a strong majority of mark 2, which seems logical, as interviewees recognize it as logical, but not precise enough. For iLCD and InfoMap, for around 60% of people the score was a 4 or a 5, therefore a good mark. (As a reminder, a score of 4 was explicitly labelled as ”The solution is good and logical, but there are a few mistakes”). If more people considered InfoMap as perfect (score 5), iLCD has significantly less bad results (score 1-2). The most interesting observation is maybe about CFinder. We can observe that this algorithm has already less 4 and 5 marks than the two other ones, but the surprising thing is that it has more marks of 1 –corresponding to incoherent results– than the connected components. What users observed is that sometimes, on some profiles, the algorithm seems to fail, and yields strange results, in which very different communities are aggregated for example, or a large community is split into several ones. It is actually known that CFinder might not be robust to some network configurations. For example, two large communities with a single clique of nodes of size k in common will be aggregated by the algorithm.

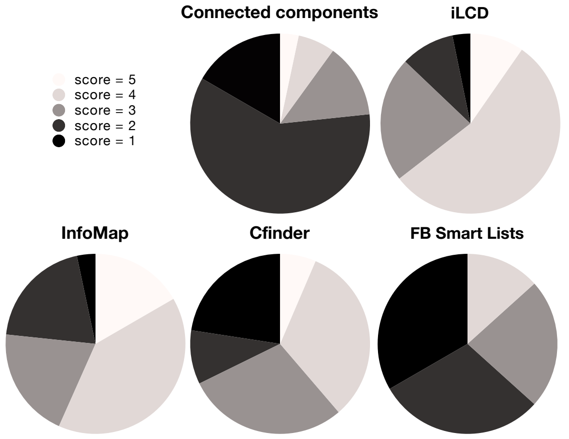

Figure 4: Score distribution by algorithm. Light colors represent good marks (4 and 5) while dark colors represent low marks (1 and 2).

We now know that the two best algorithms have a ratio of good marks (4 or 5) around 60%, but we wanted to know if they were having the good marks on the same networks or on different ones. To do so, we computed the percent of subjects that gave at least one mark of 4 or more, and we obtained 90%. This means that there is not a distinction between easy networks, on which all algorithms perform well, and hard ones on which none is efficient. In 90% of the case, at least one of the algorithms gave a good result, but frequently, only one of them do so.

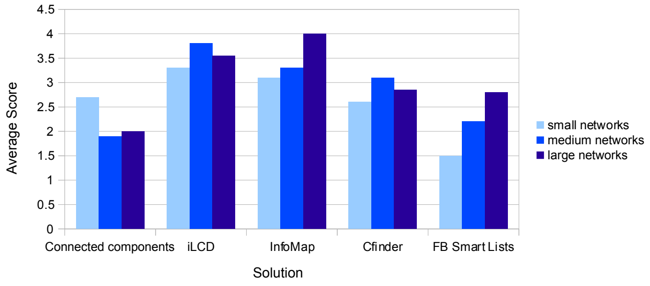

Figure 5: Efficiency of the different solutions over networks of different sizes.

Then, we were interested in seeing if some algorithms were more efficient on small or large networks. We split our data into three groups: a) users with less than 100 friends, b) between 100 and 200 friends and c) above 200 friends, which contains approximately the same number of users. Figure 5 display the results. We can observe that there is no strong correlation between the number of users and the performance of the algorithms for the 3 best methods. They do not seem to work better or worst on large networks than on smaller ones, and there is no algorithm that seems more reliable on certain type of networks. There might be some differences for the other two: the connected component analysis seems to work better on smaller networks, which is not surprising. Facebook solution, on the other hand, is getting better marks while the number of friends increase. But this result is biased by the fact that most people having defined their own lists on Facebook have a lot of friends, and they tend logically to give better marks to this solution.

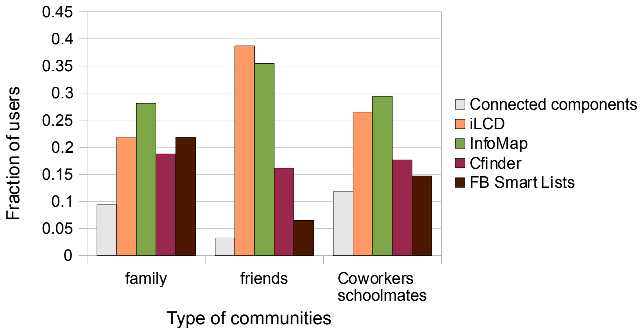

Figure 6: This chart represents how well each solution detect each kind of community. The higher a bar, the more often the corresponding algorithm has been chosen as the most efficient for the corresponding kind of community.

In another set of questions, we ask to subjects to evaluate which algorithms have identified the best their family, their friends, and their professional/scholar relations. It is first interesting to notice that subjects choose the same algorithm for the three aspects in only 33% of the cases. It means that if no solution is perfect, they are complementary. On figure 4.1.2, we show how each algorithm performs for each aspect. If the results for friendship and professional relations do not provide any surprise, with a large preference for InfoMap and iLCD, it is more contrasted for the family. Most notably, the Facebook lists seems to be as efficient as the other ones on this aspect. We can make some assumptions about that: first, families as they appear as a network in our case are not much likely to match with what the user will consider as his family. The mother's family and the father's family, for example, are usually weakly linked. But even inside one half of the family, there are frequently a high proportion of missing links on Facebook compared to what one would expect. However, Facebook gives the possibility to explicitly designate a contact as a kin, for example, sister, brother, mother, and so on and so forth. Facebook smart lists are using this to create a quite reliable family group, which corresponds more to the idea of family than what the network analysis could yield.

4.1.3 Community naming

We also asked users to evaluate the name we automatically extracted for the communities. More precisely, we asked users to evaluate the percentage of the given names that were coherent with the community they were assigned to. As explained in section 3.5, we attribute up to three words, the ones we compute as the more representative, to each community. We can see the results on fig 7. We created three categories, which we called Good, Correct and Poor, which correspond to a percentage of correct names of respectively 80%-100%, 50%-80% and 0%-50%. As we can see, for nearly half of the subjects, the users considered 80% or more of the names as relevant. This result is very positive, because it means that, in a first place, all these communities are clearly identified by the users, and though correspond to the social structure as he perceives it. On top of that, there is enough information in the users' profiles to find the common points between the persons of the group. In most cases, only a fraction of the group members filled their profiles with the relevant information, but as it is ranked as the most likely to be relevant by our method, it appears first. Less than 15% of subjects found that less than half of the groups were misnamed. In most cases, the groups are either not named (small groups with incomplete profiles), or have names that are not precise enough. For example, for a group of persons who belong to the same chess club, the name will probably be the city where the club is situated, as several persons of the groups are likely to live and/or work in this city, but this is not considered as relevant by the subjects, who expect to see the word "chess", or the club name, or any precise information. We could solve this problem by using more advanced data mining methods into the users profiles.

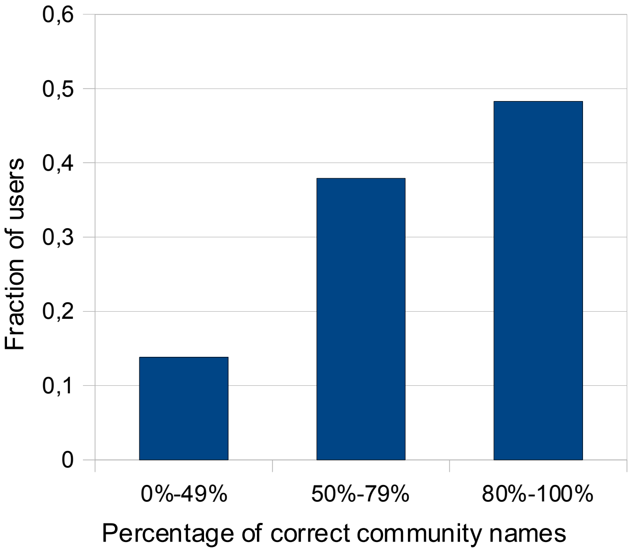

Figure 7: Repartition of the percentage of correct names. For near one half of users, more than 80% of names were corrects.

4.2 Conclusions on the proposed solution

What stands out clearly from this study is that using network analysis over one's contact list is clearly a strong improvement compared to a simple crosschecking technique, as it is done currently on Facebook. The users feedbacks were mostly positive. Most people easily recognized their communities of friends, and were searching for missing or misplaced people, which proves that the majority of the communities were correctly identified. Many subjects made a very interesting mistake: as they were not informed of the underlying mechanisms, most of them assumed that groups were made according to the names given to the communities, and therefore that the group named with their high-school name for example, was filled with their friends who have indicated so in their profile. However, when presented with the Facebook solution, which use exactly this mechanism, they mostly judged it irrelevant, with many people missing in most communities. It is also interesting to notice that one person only, in the whole experience, indicated the Facebook solution as the best one. This person was using extensively the lists on Facebook to read and publish, and logically considered his own decomposition in community as the best one.

Users gave also some negative feedbacks. Many users for example were surprised to see that some people did not appeared in some solutions. Indeed, algorithms allowing overlap (CFinder and iLCD) do not necessarily assign a community to each node. Furthermore, to keep a readable interface, we hide all communities composed of only one node. To improve this solution, it would be important to display these people in a way or another.

Several users also asked if it was possible to move persons from a community to another, to rename a community or to remove people from communities. This indicates that, of course, if people are mostly satisfied with the results, an automatic result can't be perfect and need to allow users to modify it. Such a solution, with improvements according to the weaknesses we revealed in the analysis, could have applications for users. On top of allowing a user to easily create communities from an existing large number of friends, such a tool could be used to keep up to date the list of friends. By keeping track of the new friendships of the user and his friends, and of profile updates, we could automatically propose to the user to add new friends to existing communities or to add or move a person from a community to another if needed. In order to do so, we would have to use a community detection method able to take an existing set of communities (the communities modified or defined by the user) and the new connections between the user's contacts, and to choose to modify the communities accordingly. Up to now, only iLCD is able to do so among the tested techniques, as it is originally a dynamic method.

5. CONCLUSION

In the first part of the paper, we have shown that there is actually a need for users in term of friends' groups' management solutions. In the real world, there is naturally a strong separation between groups of acquaintances, and, as social networks are not a new kind of socialization but rather an extension of the traditional social behaviors, one needs to be able to respect this separation on social networks. However, our survey revealed that there is a gap between this need and the actual usage of corresponding tools. We think that the fact that users must manually create and, even worst, update it manually is a limitation specially for people with a lot of contacts. The solution we proposed [1] is able to detect such groups of friends, corresponding to groups of friends that not only correspond to a reality, but match also with the representation that the user can have of its network. If none of the proposed methods gives very good results for all profiles, we shown that a social network analysis based method gives far better results than a profile crosschecking method, even on Facebook, which is probably the social network for which users fill the most thoroughly their profiles information. We therefore think that by enhancing one of the best proposed algorithms by using information from profile, such as family information, we could propose a tool which can be used by people to help them to create and update meaningful groups of friends, and therefore allow them to communicate and share in a more natural and controlled way.

[1] Note that you can try a modified version of the application on your own profile, from the website of the first author. The source code is also available on the same website.

6. REFERENCES

[1] A. Acquisti and R. Gross. Imagined communities: Awareness, information sharing, and privacy on the facebook. In *Privacy Enhancing Technologies*, pages 36–58. Springer, 2006.

[2] V. D. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008.

[3] D. Cardon. Le design de la visibilité. un essai de cartographie du web 2.0. *Réseaux*, 6(152):93–137, 2008.

[4] A. Casilli. Les liaisons numériques. vers une nouvelle sociabilité, 2010.

[5] R. Cazabet and F. Amblard. Simulate to detect: a multi-agent system for community detection. In *Web Intelligence and Intelligent Agent Technology (WI-IAT), 2011 IEEE/WIC/ACM International Conference on*, volume 2, pages 402–408. IEEE, 2011.

[6] C. Dwyer, S. Hiltz, and K. Passerini. Trust and privacy concern within social networking sites: A comparison of facebook and myspace. In *Proceedings of AMCIS*. Citeseer, 2007.

[7] S. Fortunato. Community detection in graphs. *Physics Reports*, 486(3-5):75–174, Feb. 2010.

[8] A. Friggeri, G. Chelius, and E. Fleury. Ego-munities, exploring socially cohesive person-based communities. *Arxiv preprint arXiv:1102.2623*, 2011.

[9] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences of the United States of America*, 99(12):7821–7826, June 2002.

[10] E. Goffman. The presentation of self in everyday life. 1959. *Garden City, NY*, 2002.

[11] M. Leguistin. Le marché de l'amour 2.0. *Esprit Critique*, 2011.

[12] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, June 2005.

[13] M. Rosvall and C. T. Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–1123, Jan. 2008.