# A User-Centric Approach for Social Data Integration and Recommendation

Yuan Wang
Department of Computer Science
University of Saskatchewan, Canada
yuw193@cs.usask.ca

Jie Zhang
School of Computer Engineering
Nanyang Technological University, Singapore
zhangj@ntu.edu.sg

Julita Vassileva
Department of Computer Science
University of Saskatchewan, Canada
jiv@cs.usask.ca

***Abstract*—It is difficult for online users to keep track of their social friendships and friends' social activities scattered across different social networking sites. We propose a user-centric approach for integrating social data from different social networking sites and allowing users to create personalized social and semantic contexts for their social data. Users can blend and group friends on different social networking sites. They can also rate friends and their activities as favourite, neutral or disliked. Our approach then provides personalized recommendations of friends' activities that may be interesting to each user. A prototype of a dashboard application called SocConnect (Social Connect) is also implemented to demonstrate the feasibility of our approach, along with evaluation on real users that confirms the appropriateness and effectiveness of our approach.**

## I. Introduction

The advent of web 2.0 technology especially social networking sites, has changed the way people communicate. Clara Shih, in her book "The Facebook Era" [1], observes that social media such as Facebook (www.facebook.com) has transformed the socio-cultural landscape – people's behavior, attitudes, interactions, and relationships. People spend more time on social networking sites than ever, and prefer communication via social networking sites over emails [2]. Every successful social networking site has its unique features. Facebook allows a large number of third party applications to build on its APIs. Twitter (www.twitter.com) offers micro-blogging and an asymmetric following relation between users. MySpace (www.myspace.com) has a large user community interested in music. LinkedIn (www.linkedin.com) focuses on career and professional networking.

Despite the diversity of social networking sites and the fact that social media enriches people's lives, current social networking sites have several significant limitations [3]. First, user-generated contents, users' online activities, and their friendships are scattered over different places. It becomes increasingly inconvenient for users to manage their social data and constantly check many sites to keep track of all recent updates. Even worse, people may have different accounts on the same social networking site. Second, social networking sites store huge amount of social data, including user profile data, friendships, events on social networking sites, interactions between users, users' media, and comments. The innovation of social networking sites has constantly increased the richness of their social data. However, these data do not have explicit context. For example, the way the word "friend" is used in Facebook does not reflect the true meaning of the word in colloquial English. On Facebook, a user's "friends" may include co-workers, college mates, and other people who she barely knows but was too polite to decline their invitations. It is thus important to have a way of distinguishing these people. Currently, for applications, context information of social data is implicit. It becomes too complex for users to make sense of the data. Third, users are often overwhelmed by the huge amount of social data, especially friends' activities (status updates). Some friends' activities are in fact not interesting to the users and should be ignored.

To address these problems, we propose a novel user-centric approach for users to organize their social data from different sources in a semantic way, that is to define social and semantic contexts of their social data. In this way, social data can be organized and filtered according to users' particular interests. More specifically, our approach is based on a new ontology of user social data and allows users to blend their friends' accounts on different social networking sites. Users can group together friends who share common characteristics despite which site these friends are from. They can also rate friends and their friends' activities as favourite, neutral or disliked. Based on the ratings, our approach also applies different machine learning techniques to learn their preferences on activities and to provide personalized recommendations of friends' activities that are interesting to them. We provide a detailed description of the underlying techniques and describe an implementation of this approach called SocConnect, a personal social dashboard. We then conduct a user study to confirm the needs of users and the appropriateness of the proposed functionalities. As evaluation of the effectiveness of our approach, we collect data from real users to show the good performance on personalized recommendations of friends' activities. In summary, by offering the above unique functionalities, our user-centric approach is thus able to provide great convenience for users to integrate, manage and view their social data scattered across different networking sites.

For the rest of the paper, we first present related work on ontology design for representing social data, applications for integrating social data, and some approaches for recommending social data in Section II. We then describe our user-centric approach, its proposed functionalities, their implementation

978-1-4244-7570-4/10/$26.00 ©2010 IEEE

details and demonstration in the prototype SocConnect in Section III. After that, we evaluate the need and appropriateness of these functionalities in Section IV. Note that it is not our goal to evaluate the interface design of SocConnect since it is not the focus of this paper. We also evaluate the performance of our approach in providing personalized recommendations of social data. Finally, we discuss the value of our work and propose directions for future work in Section V.

## II. Related Work

One important requirement for integrating social data across different social networking sites is a unified ontology to represent social data [2]. Social networking sites have their own syntaxes and terms for representing social data. The academic and open web communities have put great effort to develop standard ontologies for the representation of social data. There are several major standards, including FOAF, XFN, GUMO and Activity Stream. These standards have solid foundations; some of them have already been adopted by social networking sites and other IT companies. For example, the activity stream has been recently embraced by both Facebook and MySpace.

Another important concern in integrating social data is to keep the context of the data [3]. Otherwise, it becomes very difficult to handle the huge amount of social data and too complex for users to make sense of the data. For example, users' interactions with friends on last.fm (www.last.fm) may have different contexts from their interactions with friends on LinkedIn. On last.fm, users' interactions mostly relate to music, but on LinkedIn, the interactions are more formal and mostly relate to career development. Users and their friends on different social networking sites may also have different kinds of relationships. For example, Facebook friends are mostly people whom the user already knows [4], but users may have not met most of their Twitter friends in person. When integrating social data, the contexts should be preserved. The contexts may include the type of social bound (the semantics) of relationships (family, colleagues and friends in personal life), the common interests they share, the closeness of friendships, and the location of friends. Therefore, the ontology should also be able to allow users to express the context of social data. There are two solutions for the expression of contexts. One common way is a top-down approach that pre-defines sets of vocabularies to describe different types of social contexts. However, social contexts contain too many dimensions and too many possible variables along each dimension, of which only a few may be relevant to any given user. The process of selecting the relevant value in each dimension from a predefined ontology would be too hard for a user. The second solution is to let users themselves express social contexts by, for example grouping or rating their social data. This solution is more flexible and feasible, and we use it in our work.

There have been some attempts to create personal portals that aggregate a user's accounts on different social networking sites, for example, the Seesmic Desktop (seesmic.com), power.com and the social web browser Flock (flock.com). They allow the user to view her pages on different social networking sites in one place. In this way, the users do not have to login to many different sites to view the updates of their friends. However, these applications do not allow users to blend or group their friends from different places. They provide just a single-login interface in which users can switch between different tabs, one for each social networking site.

Bojars et al. [5] have been working on the SIOC project (Semantically-Interlinked Online Communities). This project shares similar focus with our work: social network portability and semantic web technologies. They propose the SIOC ontology, which mainly focuses on users, implicit friendship, and social contents (primarily photos and discussions) in online communities such as online forums and Weblogs where contexts of social data are not so different. In contrast, we focus mainly on developing a user-centric approach for integrating users' social data (including explicit friendship) on different social networking sites, and that allows users to organize their social data and to create their personal contexts for the social data. We also provide personalized recommendation of friends' activities that are interesting to users.

Most recommender systems use collaborative filtering [6], [7], [8] based on the sharing of user ratings. While many social networking sites deploy algorithms based on the analysis of social network structure to recommend new friends to the user, there have not been many approaches to recommend contents on social sites. One such approach is SoNARS. It takes a hybrid approach, combining results from collaborative filtering and content-based algorithms [9]. Dave Briccetti developed a Twitter desktop client application called TalkingPuffin (talkingpuffin.org). It allows users to remove "noise" (uninteresting updates) by manually muting users, retweets from specific users or certain applications. Currently, we focus on automatically providing recommendations of social networking activities mainly based on the features of the activities.

## III. Our User-Centric Approach

In this section, we first present our ontology for representing user's social data. We then describe in details the proposed functionalities of our user-centric approach and the implementation details of them in a prototype SocConnect. We also show some screenshots of the prototype.

### A. *Ontology for User's Social Data*

Social data is inherently "URI-based". Almost every piece of social data has its URI (Unique Resource Identifier). For example, a friend on Facebook has a profile URL, each Twitter update has a permanent address, and each Flickr (www.flickr.com) photo has its URL. This makes social data easy to be interlinked. A design of ontology for representing social data can easily take advantage of this feature of social data. Another feature of social data is that it incrementally changes, such as adding a new friend on a social networking site from time to time, friends' updates, and commenting on others' updates. These changes need to be synchronized. In our ontology design, we separate users' identities from their profiles and activities, inspired by the traditional software design principle "separating changes from stable elements". And, every user profile has a date stamp associated with it.

Our ontology design is also inspired by the standard of Activity Stream. The philosophy behind Activity Stream is that the essential elements of social networking sites include actors and their activities. Every user is an actor; every movement of an actor is an activity, such as adding a new friend, publishing a new blog article, and commenting on others' articles. Each activity has a type, such as Twitter update, Twitter retweet, sharing a link or a Facebook photo. The type of an activity represents the feature of this activity.

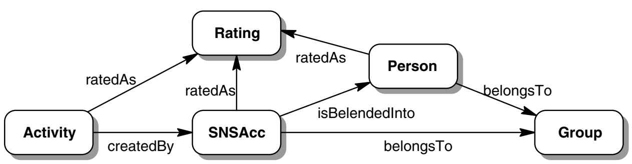


Fig. 1. Ontology for Social Data

Our proposed ontology is presented in Figure 1. There are five classes in the ontology: social networking site account (SNSAccount), integrated account (person), activity, raing, and group. "SNSAccount" represents a user account on a social networking site, e.g. Facebook, Twitter, and MySpace. "Person" represents a person who holds one or more social networking site accounts. For example, a user on Facebook may also have a Twitter account. These two SNSAccounts can be blended together. The word "person" may not be the best choice to describe the concept. For example, a user may follow CNN on Twitter, and is also a fan of CNN's Facebook page. These two sources can be integrated together as one "person" even though it is not a human being. Therefore, the class "person" actually stands for an integrated channel for multiple social data sources. This data can be converted into the FOAF format, and vice versa. "Activity" represents generic information about activities appearing on social networking sites. Activities can be user updates, events like a new friend added by the user, or a new third party application used by the user. This data can be converted into the activity stream format, and vice versa. "Rating" represents a user-generated interest level (favourite, neutral or disliked). Ratings are used to represent users' own preferences on social data. "Group" represents a user-defined group for keeping friends together. A member of a group can be a SNSAccount or a Person.

The classes are interlinked to each other. Each activity is created by a user's social networking site account SNSAccount. A person may have a set of blended SNSAccounts. A group may contain a number of persons and SNSAccounts as its members. Each activity, SNSAccount or person may be rated by some specific rating. Note that the Activity class is the core of this domain. Each activity has a SNSAccount as its actor. Activities of users or their friends incrementally fill social networks with contents. Social networking sites are essential sources of activity streams. Users and their friends are the actors of the activities.

Figure 2 provides a simplified version of user description in the XML format of this ontology.[1] This ontology not only supports collecting social data (friends and activities) from different social sites, but also allows users to create new data, for example rating friends' activities.

```
<User username='Yuan'>
  <snsAccount source="facebook" id="1234"/>
  <snsAccount source="twitter" id="Yuan_W" / >
  <friends>
    <SNSAccount source="facebook" id="3434">
      <profile date="2009-10-12" name="Jam" profile-picture="www.tinyurl.com/4er" />
      <rating>favourite</rating>
    </SNSAccount >
    <person>
      <SNSAccount source="twitter" id="1212" name="Tom"/>
      <SNSAccount  source="facebook" id="3435" name="Thomson">
    </person>
  </friends>
  <activities>
     <activity source="facebook" id="1777" actor="1212" message="my facebook update"
          createAt="2009-10-01" lastModified="2009-10-02" >
          <rating>favourite</rating>
     </activity/>
  </activities>
</User>
```

Fig. 2. Description of a User

## B. Functionalities

Our user-centric approach is based on the ontology presented in the previous section. In this approach, we propose three categories of functionalities: first, connecting different social networking sites and retrieving users' social data; second, allowing users to manage their friends; third, filtering social data by providing personalized recommendations of social data based on the ratings provided by users, and by allowing users to browse social data based on friend groups.

The first functional category, "loading social data" from different social networking sites retrieves information about users' friends and the friends' activities on these sites.

The second functional category, "managing friends" contains two functions: blending friends and grouping friends. In most cases, there is some level of overlap between the sets of a user's friends on different social networking sites. Our approach allows the user to merge the different accounts of a friend across the sites, to create a single friend. This function is a unique feature of our user-centric approach [11]. The friend can have different user accounts on different sites, but the user knows that they refer to the same person (something that no data mining algorithms can find out accurately). It is up to the user to create the mapping between her friend's accounts across different sites and assign an integrated account to represent the same friend. In this way, the user can have an integrated view of all activities of this friend, despite which social networking sites the activities come from.

The second function in the "managing friends" category is to group friends. Users can put their friends, both individual social networking site accounts and "integrated" accounts, into groups. This function allows users to express the contexts of friendships, which are the shared characteristics or interests between friends. For example, a user John who is a graduate

[1] We also designed RDF representation based on FOAF and SCOT [10].

student in Computer Science has a friend, Ben. Ben is John's buddy from a scuba-diving club, and he is also a computer scientist. Ben and John are both interested in Erlang programming and often share their findings and ideas using Twitter. They use Facebook to share their diving pictures, news about diving club events, and general news about their lives. In our approach, John will first map the two Bens he knows from Twitter and from Facebook. Next, he will define one group for his diving friends and add Ben (the Facebook Ben) into this group. He will also define an "Erlang" group and add Ben (the Twitter Ben) into it. John has another friend Vivian, she talks about Erlang programming both on Twitter and Facebook, so John can blend her accounts on these networking sites and then add the integrated person into the "Erlang" group.

The third functional category, "filtering social data" also has two functions. Users are allowed to rate social data. When new social data is retrieved, users will be provided with the recommendations about whether the new social data is interesting to them. In this way, non-interesting data can be filtered out. Another function is to allow users to browse social data based on groups. Users can view the activities of the members in the groups which they are interested in. The number of members and the amount of their social data in a group can be potentially large. It is thus still necessary to have the function of providing recommendations of social data within a group. The next section describes how these functionalities are accomplished in a prototype called SocConnect.

### C. Implementation Details

A prototype social dashboard application, called SocConnect is implemented using Adobe Integrated Runtime (www.adobe.com/products/air) to demonstrate the feasibility of the approach. As its name indicates, it connects to different social networking sites. For example, we use Twitter's "*friends show*" and "*timeline*" methods to retrieve users' friends information and their activities on Twitter. We also use Facebook's stream API for loading social data on Facebook. The retrieved data about friends and their activities from Twitter and Facebook are translated into our generic ontology that combines FOAF and Activity Stream standards together.

When a user blends a friend's SNSAccounts, SocConnect creates an instance of the Person class, and adds these SNSAccounts into the instance. The activities associated with each account link to the Person instance. When a user defines a group and adds SNSAccounts and persons into the group, SocConnect creates an instance of Group, and adds these accounts and persons into the instance. The activities associated with SNSAccounts and persons link to the Group instance. When a user rates a friend or one of the friend's activities, the rating will be added into the instance of the SNSAccount, Person or Activity class respectively.

Through our SocConnect interface, users directly express their preferences on activities by rating activities as favourite, neutral or disliked activities. Based on the ratings, SocConnect can learn users' preferences and predict whether they will be interested in new similar activities from friends. Machine learning techniques are often used for learning and prediction. SocConnect applies the classic techniques of Decision Trees, Support Vector Machine [12], Naive Bayes, Bayesian Networks, and Radial Basis Functions [13]. In brief, decision tree learning is one of the most widely used techniques to produce discrete prediction about whether a user will find an activity interesting. It classifies an instance into multiple categories. Naive Bayes Classifier and Bayesian Belief Networks are the two commonly used Bayesian learning techniques. The method of Radial Basis Functions belongs to the category of instance-based learning to predict a real-valued function. Support Vector Machines have been shown promising performance in classification problems. The implementation of these techniques bases Weka 3.7.0. The performance of these techniques on learning users' preferences on their social network activities will be presented and compared in Section IV. The one providing the best performance will be used by our system.

TABLE I
FEATURES OF ACTIVITIES FOR LEARNING

| **Features** | **A Set of Possible Values** |
|---|---|
| Actor | actor's SNS account ID |
| Actor Type | favourite; neutral; disliked |
| Activity Type | upload album; share link; upload a photo; status upload; use application; upload video; reply; twitter retweet; etc |
| Source | Facebook; Twitter; etc |
| Application | foursquare; FarmVille; etc |
| Rating | favourite, neutral, disliked |

To work with the above learning techniques, an activity needs to be represented by a set of features. Table I summarizes a list of relevant features and some of their possible values. Each activity has an actor (creator). SocConnect allows a user to rate friends into as favourite, neutral or disliked friends. Using these two features, we will be able to learn whether a user tends to be always interested in some particular friends' activities or activities from a particular type of friends. As discussed in Section III-A, each activity has a type. We also take into account the sources which activities come from, such as Facebook and Twitter, since often users have a particular purpose for which they predominantly use a given social networking site, e.g. Facebook for fun, Twitter for work-related updates. From this feature, we can find out whether a user is only interested in activities from particular social networking sites source. Different applications used to generate those activities are also useful to consider. For example, if a user's friend plays "MafiaWars" on Facebook but this user does not, the status updates generated from the "MafiaWars" application may be annoying to the user. We leave out the textual content of activities. One reason is that many activities, such as video uploads, do not have any textual content. Another reason is that activities may contain non-Latin language characters and the current meta-data of activities cannot reflect which language the actor is using, which makes text analysis difficult and expensive. After learning from a user-annotated list of activities from his or her friends, each of which is represented by a set of the feature values, our system is able to predict whether a new activity from a friend will be considered as "favourite", "neutral" or "disliked" by the user. The disliked activities will be filtered out.

### D. Demonstration

We provide several screenshots to demonstrate the user interface of SocConnect. This interface is an early prototype implementing the main functionalities proposed of our approach rather than the ultimate interface for the dashboard application. We use Facebook and Twitter for the purpose of demonstration. Suppose that a user Jane has accounts on both Facebook and Twitter. SocConnect retrieves Jane's social data on these two sites. The social data of her friends can then be managed and filtered by her SocConnect dashboard based on her personal needs or interests. We step through an example to show more specifically what Jane can do with the application. The social networking site accounts of the actual users in the screenshots are blacked out to protect their privacy.

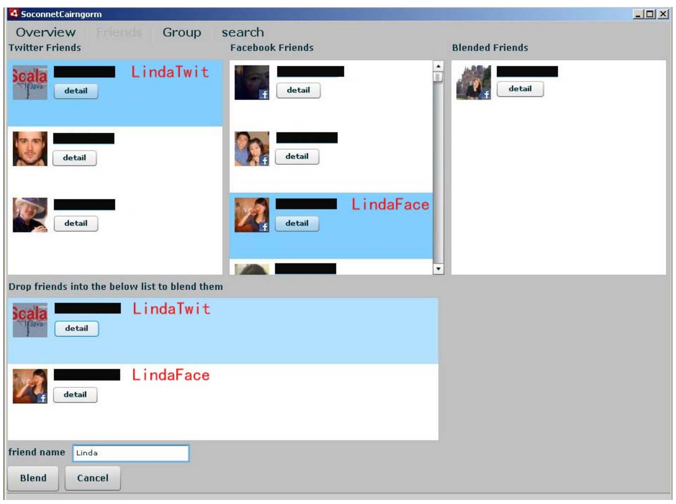


Fig. 3. Blending Friends

Jane can use SocConnect to blend her friends who have social networking site accounts on both Facebook and Twitter. As shown in Figure 3, there are three lists in the upper part. The left list contains Jane's friends on Twitter and the middle one contains her friends on Facebook. Jane drags her friend Linda's Twitter account "LindaTwit" from the left list and Linda's Facebook account "LindaFace" from the middle list to the lower list. By clicking the "Blend" button shown in the bottom of the figure, Linda's accounts in the lower list are joined into a "blended" person. Jane gives a name "Linda" for the blended person. The third list in the upper-right part of the screen shows the list of all Jane's "blended" persons. Linda will be added to the list.

Jane can also use SocConnect to group her friends together. As shown in Figure 4, the interface for this function is similar to the interface for blending friends. To add members into a group, Jane can drag her friends' accounts from the three lists in the upper part of the figure and drop them into the list in the lower part. She drags her friends in New Jersey into the lower list, including John and Bob from the Twitter list and Amy from the Facebook list. She also drags the blended person Linda into this list from the list of blended persons. She gives the name "friends@NJ" to the group and clicks the button of "Create a new group" in the bottom of the screen. A new group is then created for Jane, and the list of Jane's

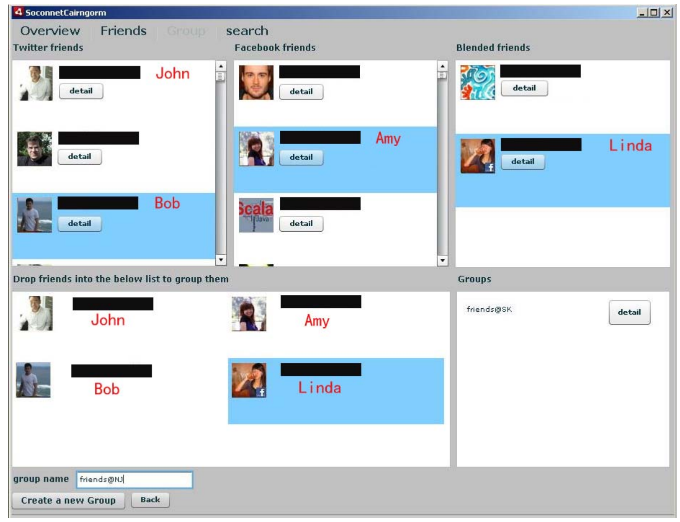


Fig. 4. Grouping Friends

groups is shown in the right most list in the lower-right part of the screen. A user can also put her friends in different groups, e.g. John can be both a member of Jane's "friends@NJ" group and her "friends@SK" group.

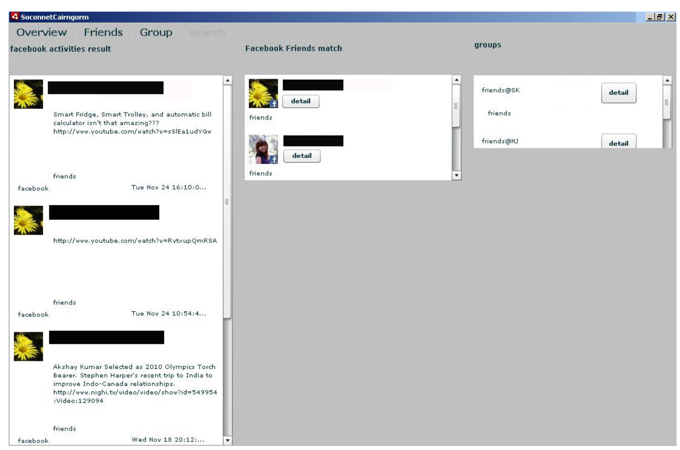


Fig. 5. Filtering Social Data

The function of grouping friends provides a flexible way for users to organize their friends by contexts. It also allows users to filter only social data from the members of a particular group. For example, Jane can check news from friends@NJ by clicking the group name listed in the right most list of Figure 5 marked by "Groups". A list of the members in this group will appear in the middle list, and the updates from these members will appear in the left most list. In addition, the function of providing personalized recommendations of friends' activities predicts whether certain updates are interesting to Jane. Only the updates that are predicted to be interesting to Jane will be displayed in the left most list. Neutral ones can be displayed upon request. Disliked ones are filtered out.

## IV. Evaluation

We first conducted a user study to verify the need for the proposed functionalities and their appropriateness. Questions were asked to evaluate the necessity of the main functionalities, including blending friends, grouping friends, and filtering social data. For evaluating the necessity of the function of blending friends, the following questions were asked: 1) Do you keep friends on different social networking sites for different purposes? 2) Do you have some friends who have accounts on several social networking sites? If so, how many roughly? 3) Have those friends been active on these sites? 4) Do they mostly have identical activities on these sites? 5) If they mostly have identical activities on these sites, do you want to view their activities in one place?

The function of blending friends is necessary only if users have some friends who have accounts on different (at least two) social networking sites. The positive answer of Question 2 (Q2) is then the prerequisite of having this function. But, even if a user has the same friends on different sites, the user may still not want to blend these friends if she keeps friends on different social sites for different purposes or contexts. Therefore, the negative answer of Q1 is also the prerequisite. To argue that the function is actually necessary and useful, users' friends have to be active on different sites (Q3) and users should feel that it is valuable to view friends' identical activities in one place (Q4 and Q5).

The questions related to the necessity of having the grouping friends function were as follows: 6) Some social networking sites allow you to put some friends into a list (a group). Have you ever used this function? 7) Do you have some friends who share similar interests, preferences, or demographic information, or do some activities together? 8) Do you want to create a group for these friends? 9) Do you also want to include in groups some friends on different sites?

Some social networking sites (i.e. Facebook and Twitter) provide the grouping function. If users have already made good use of this function (Q6), this becomes positive indication for the function of grouping friends. However, this function is fairly new to the social sites. It is likely that some users have not paid much attention to this function. Our other questions also provide estimation for the necessity of the function. It is very likely that users have friends who share some commonalities (Q7) if the users have many friends. We still ask this question, in order to guide subjects to be focused on their friends who are in common for the next question (Q8). Question Q9 is related to the special functionality that we propose which integrates users' friends (who are on different social sites) into one place. This allows to put those friends in one group, which is impossible via a single social networking site such as Facebook or Twitter.

For the function of filtering social data, the following questions were asked: 10) Have you ever had difficulty in browsing through your friends' updates, and have you been overwhelmed? 11) Do you want to view your friends' activities (updates) by groups? 12) Do you want to organize friends and their updates by tagging? If a user has many friends, the answer to Question 10 is likely positive. It is necessary to have the function of filtering social data, especially in the case where the user's friends on different social sites are now gathered in one place (as in our application). Question Q11 provides indication whether the function of grouping friends will be helpful for users' navigation of activities. Question Q12 investigates users' preferences about tagging friends and their activities, instead of simply rating friends and their activities.

Our study involved 16 subjects (all students). Table II summarizes the demographic information about these subjects. We can see that they are distributed over both gender (Male or Female) and major (Computer Science or Non-CS).

TABLE II
Demographic Information about Subjects

| Subjects | Computer Science | Non-CS | Total |
|---|---|---|---|
| Female | 5 | 3 | 8 |
| Male | 5 | 3 | 8 |
| Total | 10 | 6 | 16 |

For evaluating the necessity of the blending friends function proposed by our approach, Questions 1-5 were asked. Figure 6 is the summary of the number of each subject's friends who have user accounts on more than two social sites (Q2). As can be seen, only two subjects do not have such friends. More than a half subjects have at least 7 such friends. Several subjects (25% of all subjects) have more than 20 such friends. This result suggests a strong need for the function of blending friends.

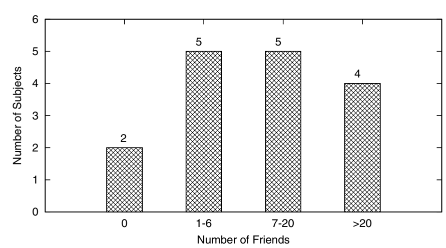


Fig. 6. Number of Friends Having Accounts on More than Two Sites

The summary of the answers for Questions 1 and 3-5 is presented in Table III.[2] The results of Questions 3-5 indicate the subjects' strong desire for the function of blending friends. Note that Q4 is overly strict. In fact, three subjects provided the answer of "50% similar and 50% identical". Within these three subjects, two subjects still provided positive answers to Q5 and only one subject was not sure whether the function

[2]For the two subjects who do not have friends with accounts on more than two social sites as pointed in Figure 6, we assume that they do not support the function of blending friends and will be negative for our questions 1 and 3-5.

is important. The result of Q1 is not significant even though most of the subjects do not keep friends on different social networking sites for different purposes. The reason that almost a half of the subjects keep friends on different sites for different purposes is because many of them also use social sites that are most popular in their own country, for example, Orkut (www.orkut.com) of India and Xiaonei (www.renren.com) of China. They often keep friends in their own countries on these sites and friends in other countries on Facebook or Twitter. For most of these subjects, some of their friends still have accounts on different sites (see the result of Q2 in Figure 6). For example, their friends may have accounts on both Orkut and Facebook. These subjects want to view their friends activities of those friends in one place (see the result of Q5 in Table III).

TABLE III
RESULTS RELATED TO BLENDING FRIENDS FUNCTION

| | Yes | | No | |
|---|---|---|---|---|
| Questions | Num | Percent | Num | Percent |
| Q1 | 7 | 43.75% | 9 | 54.25% |
| Q3 | 13 | 81.25% | 3 | 18.75% |
| Q4 | 11 | 68.75% | 5 | 31.25% |
| Q5 | 12 | 75% | 4 | 25% |

As expected, from the subjects' answers to Q6, only three subjects have used the new function of grouping friends offered by Facebook or Twitter. All these three subjects provided positive answers to Q7, Q8 and Q9, which indicates that they like the function of grouping friends and think that the function is necessary. The subjects' answers to Q7, Q8 and Q9 related to the function of grouping friends were also very positive to support this function, as can be seen from Table IV. Only one subject (out of 16) was consistently against this function.

TABLE IV
RESULTS RELATED TO GROUPING FRIENDS FUNCTION

| | Yes | | No | |
|---|---|---|---|---|
| Questions | Num | Percent | Num | Percent |
| Q7 | 15 | 93.75% | 1 | 6.25% |
| Q8 | 15 | 93.75% | 1 | 6.25% |
| Q9 | 14 | 87.5% | 2 | 12.5% |

The subjects' answers to Q11 are summarized in Table V. 81.25% of the subjects support this function of allowing them to view their friends' activities by groups. They think that this function of filtering social data by groups will provide much convenience for reading friends' updates. This result is further supported by the answers to Q10 that most of the subjects have been overwhelmed by a number of their friends' updates in one social networking site. The number of updates will increase significantly when the friends' accounts across different social sites are integrated by our SocConnect. Thus, it is also necessary to provide recommendations of friends' updates that are interesting to the users according to their preferences, and filter out the ones that are not interesting. The result of Q12 (whether to tag friends' activities) is not so significant. Tagging requires effort. The subjects were not sure whether they want to spend much time on tagging. Some subjects also feel that not many updates are important to be tagged. These results actually provide strong reasons for our approach of allowing users to simply rate friends and their activities, which minimizes effort required from users.

TABLE V
RESULTS RELATED TO FILTERING SOCIAL DATA

| | Yes | | No | |
|---|---|---|---|---|
| Questions | Num | Percent | Num | Percent |
| Q10 | 11 | 68.75% | 5 | 31.25% |
| Q11 | 13 | 81.25% | 3 | 18.75% |
| Q12 | 9 | 54.25% | 7 | 43.75% |

We also carried out experiments to confirm the performance of our function for personalized recommendation of friends' updates, by evaluating the five machine learning techniques for predicting user preferences on social activities. Twelve subjects were involved in our evaluation. Five of them are from Saskatoon, Canada, and the other seven are from New Jersey, USA. A half of them are students and the other half are workers. We collected from the subjects the recent Facebook and Twitter activities from their friends. Ten of the subjects are experienced users of Facebook and Twitter. For each of these subjects, we collected 100 recent activities of friends. The other two subjects are relatively new users of Facebook and Twitter. For each of them, we collected around 50 recent activities of friends. We asked all subjects to rate their friends and activities. On average, they rated 38% of their friends as favourite or disliked friends and 45% of the activities as favourite or disliked, thus representing quite a diverse data sample.

A 10-fold cross validation was performed on the collected data from each subject, and the average performance of the machine learning techniques over the activities of all subjects are reported in Figure 7. Although the performance difference among these techniques is not very significant, support vector machine (SVM) provides the best performance, and it correctly classifies 74.1% of instances in the testing data. RBF performs the worst (70%). The performance of Naive Bayes and that of Bayesian Belief Network are about the same (around 72.6%). Decision Tree performs a little better (71.4%) than RBF.

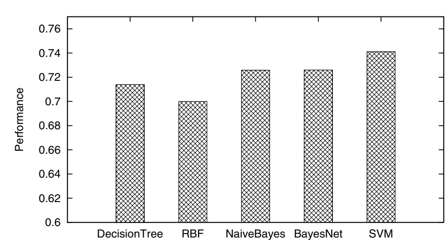


Fig. 7. Performance Comparison among Machine Learning Techniques

By looking closely into the predicted results, we found that many instances were misclassified by only one interest level, i.e. from "favourite" to "neutral" or from "disliked" to "neutral" and vice versa. We consider these as smaller mistakes. We summarize in Figure 8 the percentage of more serious misclassification from "favourite" to 'disliked" and vice versa. We can see that only a very few (less than 3%) activities have been misclassified from "favourite" to "disliked" and vice versa. SVM consistently shows its best performance in this case. Overall, the experimental results confirm the good performance of machine learning techniques in learning social networking users' preferences on their friends' activities. SVM is particularly recommended in this context.

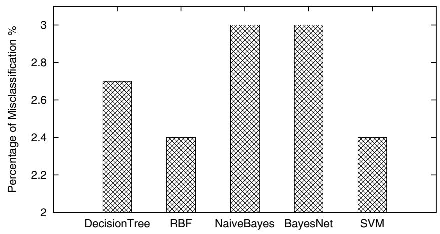


Fig. 8. Percentage of More Serious Misclassification

We also performed the validation on only 50% of collected data. More specifically, for each subject, we randomly selected 50% of collected instances. For each half of the data, we performed the same 10-fold cross validation to test the performance of the machine learning techniques. We repeated this process for 10 times to get the average performance when using only 50% of collected data. Results shown in Figure 9 indicate that the performance when using 50% of data is consistently lower than that when using all data for the five machine learning techniques. This implies that the performance of personalized recommendation on social activities can be much improved when more data is collected from users, as users continuously use our system.

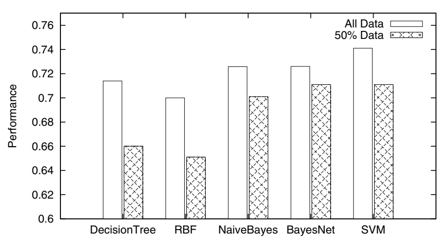


Fig. 9. Performance When Using All Data verses 50% Data

## V. Discussion and Future Work

In this work, we designed a generic ontology to describe social data from different social networking sites. Based on this ontology, we developed a user-centric approach for integration of social data from different sites and implemented it in a personal dashboard application called "SocConnect". Unlike other integration applications such as power.com and Flock, our approach allows users to integrate and organize social data coming from different sites in a semantic way. It allows users to easily define their personal contexts of social data. For example, if a user has a friend both on Facebook and Twitter, SocConnect allows the user to associate together the feeds of this friend from both social networking sites, so that the user can browse this friend's activities independently from where (in which site) the activities are posted. It also allows users to define and express the semantics of their relationships by ratings. Users can also define groups and add their friends into these groups. Users of our application can selectively view the social data posted by members of a particular group. They are also provided with personalized recommendations of social data that is interesting to them. Results of our user study indicate strong support for the functionalities proposed in our approach. Evaluation on real user data also confirms the sufficient performance for the personalized recommendations of social data. In summary, our approach provides effective social data integration and recommendation.

For future work, we are interested in exploring more deeply the relative importance of different features of social networking activities, to further improve the performance of personalized recommendation of activities. Other features that may be worth looking at include textual content of activities and the targeted friends of friends in activities. We will also look into the sharing of ratings of activities among users of SocConnect. In this case, Collaborative Filtering [6] will be used for predicting whether an activity is interesting to a user based on other users' ratings for the activity. We will also improve the interface of SocConnect and conduct user studies on the improved user interface to evaluate its usability.

## References


[1] C. Shih, *The Facebook Era: Tapping Online Social Networks to Build Better Products, Reach New Audiences, and Sell More Stuff*. Prentice Hall PTR, 2009.

[2] M. Chisari, "The future of social networking," in *Proceedings of the W3C Workshop on the Future of Social Networking*, 2009.

[3] G. Erétéo, M. Buffa, F. Gandon, M. Leitzelman, and F. Limpens, "Leveraging social data with semantics," in *Proceedings of the W3C Workshop on the Future of Social Networking*, 2009.

[4] C. Lampe, N. Ellison, and C. Steinfield, "A face(book) in the crowd: Social searching vs. social browsing," in *Proceedings of the ACM Special Interest Group on Computer-Supported Cooperative Work*, 2006.

[5] U. Bojars, A. Passant, J. Breslin, and S. Decker, "Social network and data portability using semantic web technologies," in *Proceddings of the Workshop on Social Aspects of the Web*, 2008.

[6] P. Resnick, N. Lacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: an open architecture for collaborative filtering of netnews," in *Proceddings of the ACM conference on Computer supported cooperative work*, 1994.

[7] H. Kautz, B. Selman, and M. Shah, "Referral web: combining social networks and collaborative filtering," *Communications of the ACM*, vol. 40, no. 3, pp. 63–65, 1997.

[8] N. Goo, J. B. Schafer, J. A. Konstan, A. Borchers, B. Sarwar, J. Herlocker, , and J. Riedl, "Combining collaborative filtering with personal agents for better recommendations," in *Proceddings of the sixteenth national conference on Artificial intelligence*, 1999.

[9] F. Carmagnola, F. Vernero, and P. Grillo, "Sonars: A social networks-based algorithm for social recommender systems," in *Proceddings of the 17th International Conference on User Modeling, Adaptation, and Personalization*, 2009.

[10] H.-L. Kim, J. G. Breslin, S.-K. Yang, and H.-G. Kim, "Social semantic cloud of tag: Semantic model for social tagging," in *Proceddings of the KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications*, 2008.

[11] C. A. Yeung, I. Liccardi, K. Lu, O. Seneviratne, and T. Berners-Lee, "Decentralization: The future of online social networking," in *Proceedings of the W3C Workshop on the Future of Social Networking*, 2009.

[12] J. C. Platt, "Fast training of support vector machines using sequential minimal optimization," in *Advances in Kernel Methods: Support Vector Learning*, B. Schoelkopf, C. Burges, and A. Smola, Eds. MIT Press, 1999.

[13] T. M. Mitchell, *Machine Learning*. McGraw-Hill, 1997.