Connecting Corresponding Identities across Communities

Reza Zafarani and Huan Liu
Department of Computer Science and Engineering
Arizona State University
{reza, huanliu}@asu.edu

Abstract

One of the most interesting challenges in the area of social computing and social media analysis is the so-called community analysis. A well known barrier in cross-community (multiple website) analysis is the disconnectedness of these websites. In this paper, our aim is to provide evidence on the existence of a mapping among identities across multiple communities, providing a method for connecting these websites. Our studies have shown that simple, yet effective approaches, which leverage social media's collective patterns can be utilized to find such a mapping. The employed methods successfully reveal this mapping with 66% accuracy.

Introduction

Community analysis has been an interesting problem in the recent developments of Data Mining and Social Media Analysis (Wasserman & Faust 1994). Here, a *community* refers to a specific social media website (e.g., *StumbleUpon*). It is worth mentioning that the current research seeks to analyze communities by means of different techniques such as Link Analysis and Opinion Mining (Flake, Lawrence, & Giles 2000; Hu & Liu 2004). However, in most cases, if not all, analyses are restricted to a single community. A major problem when dealing with any kind of cross-community analysis is the disconnectedness of these communities. The missing element is the connectivity among users in different communities, which is an essential factor in any link analysis algorithm. This is due to the unrevealing nature of the web and the fact that most communities preserve the anonymity of users by allowing them to freely select usernames instead of their real identities and the fact that different websites employ different username and authentication systems. Furthermore, communities rarely share Single-Sign-On procedures, where users can logon to different communities using a single username (e.g., as in *Orkut* and *YouTube*). Nevertheless, if there exists a mapping between usernames across different communities and the real identities behind them, then connecting communities across the web becomes a straightforward task. Can we find this mapping? In this paper, we provide evidence on the existence of this mapping, and demonstrate a step-by-step procedure to discover corresponding identities across communities. We first formally present the corresponding identity elicitation problem, then present empirical observations regarding the behavior of users on the web, next discuss our proposed method for identifying corresponding identities, followed by our experimental results and conclusions.

Cross-Community Corresponding Identity Elicitation Problem

Many properties of web communities can be employed to elicit connections among them. Usernames are one of them. Another is E-mail addresses. The uniqueness of E-mail addresses can serve as a universal identifier of individuals across different communities. However, email addresses may not be as much available as usernames. Therefore, we focus on employing usernames. We formalize the problem of using usernames as a community-linkage tool below.

Let μ_i represent an active individual in the cyberspace. Let C represent the set of all communities and $c_j \in C$ represent a single community. Let $C_{\mu_i} \subset C$ denote the set of all communities in which user μ_i has a username. We denote the set of all active users in community c_j as Λ_{c_j}. Let $U(\mu_i, c_j), c_j \in C_{\mu_i}$ represent the username user μ_i has in community c_j and let U^{-1} represent the inverse function ($username \rightarrow user$) such that $U^{-1}(U(\mu_i, c_j), c_j) = \mu_i$. Furthermore, a *username-username* pair $< u_1, u_2 >$ for some user μ_i and communities c_j and c_k, such that $\mu_i \in \Lambda_{c_j},\ \mu_i \in \Lambda_{c_k}$ is defined as follows:

$$< u_1, u_2 >:\ \ U(\mu_i, c_j) = u_1,\ U(\mu_i, c_k) = u_2,$$

whereas, a *username-community* pair $< u_j, c_k >$ for some user μ_i is defined as follows:

$$< u_j, c_k >:\ \ U(\mu_i, c_k) = u_j,\ \mu_i \in \Lambda_{c_k}$$

Moreover, a *username-set* for user μ_i, Σ_{μ_i}, is defined as:

$$\Sigma_{\mu_i} = \{U(\mu_i, c_j) | c_j \in C_{\mu_i}\}$$

Similarly, a *community-username-set* for community c_j, Π_{c_j}, is defined as:

$$\Pi_{c_j} = \{U(\mu_i, c_j) | \mu_i \in \Lambda_{c_j}\}$$

Then, cross-community corresponding username elicitation can be formally stated as follows:

Definition. ***Cross-Community Corresponding Username Elicitation:*** *given a username-community pair $< u_1, c_1 >$, called base-username and base-community, and a community c_2 (target community), a solution to the cross-community corresponding username elicitation problem is a username $u_2 \in \Pi_{c_2}$, called the target-username, such that $U^{-1}(u_1, c_1) = U^{-1}(u_2, c_2)$.*

We next present some hypotheses on the relationship between usernames selected by a single person in different communities, and on some of the web phenomena regarding usernames and communities. These hypotheses are evaluated based on empirical experiments. The results from these experiments, as we will see, tend to be useful in devising our proposed method for corresponding-username extraction.

Empirical Observations

We present 7 hypotheses, each of which, if required, is formally defined and then empirically validated. The observations gathered while evaluating these hypotheses are used later on to help construct our proposed method for extracting corresponding identities in other communities. Note that in order to evaluate these hypotheses we required a sufficiently large dataset from which labeled data could be acquired. For this purpose, we have used the *BlogCatalog* (http://www.blogcatalog.com/) web community and developed a data fetching engine for this website. BlogCatalog is a comprehensive directory of blogs which not only provides useful informations about various weblogs, but also comprises different facilities for users to interact within its community. What is more interesting about BlogCatalog is that users in BlogCatalog are provided with a feature called "My Communities". This feature enables users to list their usernames in other communities. Our engine has gained advantage of this feature of BlogCatalog and has collected a large set of usernames in this community, along with their corresponding usernames in other web communities. Overall, 38,093 *username-username* pairs were gathered. Each pair consists of the username in the BlogCatalog community and the corresponding username in another community. Besides BlogCatalog, the dataset contains usernames from 36 different communities. From this dataset, the other datasets required for all our experiments were generated.

Hypotheses

Before delving into these hypotheses, we formally define some of the notations. Let $Domain(c_i)$ denote the Registered Domain Name of community c_i. Furthermore, for any Registered Domain Name d_i and for any URL URL_i, $URL_i \in d_i$ denotes that URL_i is on domain d_i. Finally, the *URL-set* of community c_i, Φ_{c_i}, is defined as follows:

$$\Phi_{c_i} = \{URL_i | URL_i \in Domain(c_i)\}$$

$\mathcal{H}_1$**: for any username u_i and community c_j s.t. $u_i \in \Pi_{c_j}$, there exist a non-empty set $S \in \Phi_{c_j}$, for which the following holds true: $\forall url \in S$, u_i is a sub-string of url.** Informally speaking, this hypothesis states that for most communities and for all usernames residing on them, there exists URLs on the community website that contain the username. These URLs are most commonly pointing to the profile/homepage of the users on that community. As an example, consider how the profile page URLs of a fictional user $test$ can be reached on some of the most popular social networking websites in Table 1. In order to empirically prove this phenomenon, we have analyzed more than 36 online community websites and surprisingly, in all 36, there exist URLs that contain the username, i.e., 100% accuracy.

MySpace	`http://www.myspace.com/`**`test`**
YouTube	`http://www.youtube.com/`**`test`**
Del.icio.us	`http://del.icio.us/`**`test`**

Table 1: Profile URLs for Popular Social Networking Webs

$\mathcal{H}_2$**: given a community c_i, it is highly probable to identify $Domain(c_i)$ using web search engines.** In order to approximate the validity of this hypothesis, we used all 36 communities available in our dataset. For each community, a Google search was performed with c_i as the query, e.g., Flickr. It was found that in all cases, the first retrieved URL was the community's Registered Domain Name ($Domain(c_i)$), i.e., perfect accuracy (100%) was achieved.

$\mathcal{H}_3$**: for any username u_i and community c_j s.t. $u_i \in \Pi_{c_j}$, it is highly probable to discover, using web search engines, a non-empty set $S \in \Phi_{c_j}$, for which the following holds true: $\forall url \in S$, u_i is a sub-string of url.** It has been empirically proven in the first hypothesis that if a user is active on some community, then there exist URLs containing his/her username on the community's domain. Given this fact, this hypothesis suggests that these URLs can be easily found on the web using web search engines. Note that if all the existing communities on the web were known, then we would have been able to simply use the pattern through which the user profile's URL is generated on that specific community (see Table 1) and then, check if this generated URL existed on the community website (e.g., no HTTP 404 error is encountered); however, a more realistic scenario is the case where we do not know anything about the URL pattern of the user-profiles and we are only provided with the community name. In this scenario, the first challenge is to find the community's Registered Domain Name (e.g. myspace.com) and then, find the URLs, such as the user's profile, which contain the username (e.g., myspace.com/**u** for username **u**). As previously discussed, given the community name, the community's Registered Domain Name can be found quite easily. We have also shown, based on $\mathcal{H}_1$, that the username exists in a non-empty set of URLs residing on the community's domain name in all cases. Hence, the task of finding this non-empty set of URLs is reduced to the task of finding URLs that not only reside on the community's domain, but also contain the username in them. This task can be easily performed using the `inurl` (Searches within URLs) and `site` (Searches within the webpages residing on some specific Registered Domain Name) features of the Google search engine (other search engines provide similar services). Another view of this hypothesis is that it analyzes the likelihood of the set of URLs containing username (e.g., user's profile) being indexed by the search engine. We analyzed more than 45,565 *username-community* pairs

$< u_i, c_j >$ for this experiment. A search on Google with "inurl:u_i site:$Domain(c_j)$" as the query was performed. Our experiments showed that in nearly 81% of the cases, at least one URL is retrieved satisfying our conditions.

$\mathcal{H}_4$: for any user μ_i, if $|\Sigma_{\mu_i}| > 1$, then for any two usernames u_1 and u_2 in Σ_{μ_i}, there is a high chance of co-occurrence of these two in search engine results. To evaluate this hypothesis, we generated 41,241 *username-username* pairs $< u_1, u_2 >$, i.e., both u_1 and u_2 belonged to the same person's *username-set*. We found using Google that usernames co-occur in nearly 68% of the situations. Since this hypothesis holds with a reasonable accuracy, we can perform a web search using one of the usernames and then perform keyword extraction on the retrieved webpages to discover the other usernames; however, though sufficiently accurate, in some cases, the retrieved URLs are many and as a direct result, keyword extraction can be quite tedious. So, we proposed another hypothesis, which deals with a somewhat more restricted version of the current one, yet can be quite useful.

$\mathcal{H}_5$: for two username-community pairs, $< u_1, c_1 >$ and $< u_2, c_2 >$ of the same user μ_i, it is sufficiently likely for u_1 to exist on webpages retrieved using popular search engines whose URLs are a member of a non-empty set $S \in \Phi_{c_2}$ and for which the following holds true: $\forall url \in S$, u_2 is a sub-string of url. This hypothesis analyzes the chance of a username of a person occurring on the webpages whose URL contain the other username (e.g., user's profile). Again, to evaluate this hypothesis, we generated 41,241 *username-username* pairs $< u_1, u_2 >$, i.e., both u_1 and u_2 belonged to the same person's *username-set*. For each pair, two separate queries were sent to Google (first username occurring on URLs containing the second username, and vice versa). These queries were in the following format: "inurl:u_1 u_2" and "inurl:u_2 u_1". We found that this hypothesis holds in nearly 38% of the situations. Likewise our previous hypothesis, and based on the results of this hypothesis, we can perform a web search using one of the usernames and then perform keyword extraction on the URLs of the webpages retrieved to discover other usernames.

$\mathcal{H}_6$: for any user μ_i, it is highly probable to have $|\Sigma_{\mu_i}| = 1$. This hypothesis states that people tend to use the same username in different communities. If this holds, then the only requirement for extracting corresponding usernames in different communities is to find a single username of an individual. In order to approximate the validity of this hypothesis, we gathered 101,179 *username-username* pairs $< u_1, u_2 >$, i.e., both u_1 and u_2 belonged to the same person's *username-set*. It turns out that users have selected the same username in 59% of the situations. Moreover, 6% *username-username* pairs are pairs for which one of the usernames is created using the other one by adding a suffix, and another 1% are the ones that are created by adding a prefix. Finally, even if the usernames are not equal or created using a prefix or suffix, there is 2% chance that they have a small Levenstein distance, also known as Edit distance, from each other (e.g., <BobLee, Bob1Lee>). So, given common prefixes/suffixes, an accuracy of around 66% is expected.

$\mathcal{H}_7$: for any user μ_i, it is highly probable to have $|\Sigma_{\mu_i}| < |C_{\mu_i}|$. This is more general than the previous hypothesis and what it states is that people tend to use one of their many usernames in different communities. If this hypothesis holds, then the requirement for extracting corresponding usernames across multiple communities is to find different usernames of a person and try each username on the community's website (e.g., check if the profile exists). In order to approximate the likelihood of this hypothesis, we evaluated this hypothesis over 36,214 usernames. It turns out that users have selected the same username as one of their many usernames in most (77%) cases. Moreover, a 5% of the usernames are created by adding suffixes to one of their other usernames, and another 1% are the ones that are created by adding a prefix. So, again, if one can find all the other usernames and popular prefixes/suffixes, then on can expect 83% accuracy.

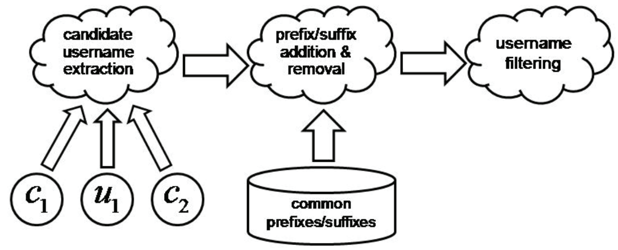

Figure 1: Corresponding Username Extraction

An Approach to Cross-Community Corresponding Username Elicitation

In this section, we overview our proposed method to identify corresponding usernames across communities. The procedure is depicted in Figure 1. The input to this process is the base username u_1, base community c_1, and the target community c_2. The procedure starts with finding a set of keywords, for which it believes can be candidates for the corresponding usernames in the target community. Then, in addition to keeping the original keywords, this set is expanded by adding/removing common prefixes and suffixes to/from its members. Note that since we have found out that *any* of the both usernames can be created by adding prefixes and suffixes ($\mathcal{H}_6$ and $\mathcal{H}_7$) to the other, hence we also remove prefixes and suffixes from these candidates. Finally, the members of this set are checked with the target community in order to filter out keywords which do not represent usernames in the target community.

As discussed previously ($\mathcal{H}_5$): *usernames appear in the URLs of the profile webpages of each other*. In *Candidate Usernames Extraction*, we use this principle to extract our username sets for each username. Given a username, based on hypothesis $\mathcal{H}_5$, we know that usernames co-occur in each other's profiles; therefore, we search for our base-username on Google hoping for it to be found on the user's target-community profile or some other profiles of the same person. Since the usernames occur in the URL ($\mathcal{H}_1$), we extract keywords from all the retrieved URLs. These keywords are preprocessed and the remaining keywords are assumed

	Del.icio.us	Digg	Flickr	Furl	Last.fm	Multiply	MyBlogLog	MySpace	Reddit	StumbleUpon	Technorati	Twitter	YouTube
Del.icio.us	1	0.68	**0.66**	0.84	0.76	0.62	**0.73**	0.47	0.9	0.72	0.78	0.76	0.58
Digg	0.7	1	0.57	0.78	0.82	0.54	0.63	0.4	0.84	0.62	0.68	0.64	0.54
Flickr	0.66	0.64	1	0.66	0.71	0.45	0.51	**0.58**	0.56	0.63	0.59	0.65	0.6
Furl	0.78	0.76	0.63	1	**0.88**	0.74	**0.73**	0.45	**0.92**	0.78	**0.82**	0.76	0.6
Last.fm	0.74	0.78	0.6	0.82	1	0.64	0.64	0.53	0.72	0.64	0.72	0.64	0.54
MyBlogLog	0.71	0.67	0.47	0.63	0.66	0.46	1	0.35	0.71	0.6	0.67	0.67	0.47
MySpace	0.57	0.56	0.54	0.61	0.57	0.49	0.56	1	0.57	0.52	0.53	0.53	0.58
Reddit	**0.84**	**0.8**	0.54	**0.86**	0.68	**0.78**	0.67	0.43	1	**0.8**	0.76	**0.77**	**0.62**
StumbleUpon	0.74	0.68	0.5	0.78	0.68	0.6	0.62	0.38	0.86	1	0.66	0.6	0.58
Technorati	0.74	0.66	0.5	0.8	0.72	0.48	0.65	0.4	0.78	0.64	1	0.66	0.58
Twitter	0.64	0.64	0.53	0.68	0.7	0.53	0.65	0.33	0.81	0.58	0.62	1	0.52
YouTube	0.58	0.6	0.58	0.6	0.68	0.52	0.55	0.56	0.67	0.6	0.68	0.62	1

to be candidate usernames. The preprocess procedure removes common words such as the protocol names, famous sub domains, index files, extensions, etc. As mentioned in hypotheses $\mathcal{H}_6$ and $\mathcal{H}_7$, after analyzing the corresponding *username-username* pairs, we found that users tend to create new usernames by adding prefixes or suffixes to their other usernames. We gathered in our data all the prefixes and suffixes employed by the users in two separate sets. We then sorted these sets based on their frequency and selected frequent prefixes and suffixes. A prefix or suffix is considered frequent, if its frequency is statistically significant. In our experiments a frequency more than 2.5σ far from the mean frequency is considered significant, where σ is the standard deviation of frequencies. Prefixes such as {*the, i, b, iam, my, free, happy, dr, x, mister, coach*}, or suffixes such as {*1, 2, s, dotcom, b, blog, 7, 07, 77, 13, a, z, 66, 0, 50, 08, com, e, art*} were commonly used in our collected dataset. The set of candidate usernames is further expanded using these prefixes and suffixes in order to generate the final set of usernames. It is also worth mentioning that by using some Google search engine features (e.g., using the * operator) the prefix/suffix list can be further expanded. Finally, given this set of candidate usernames, in order to filter out usernames, we check for the existence of these usernames on the URLs that reside in the target community domain. Note that we are already sure ($\mathcal{H}_1$) that there exist URLs which contain these username. For each candidate username u_i, this procedure is performed by a web-search on Google with "inurl:u_i site:$Domain(c_2)$", where c_2 is the target community. If the quantity of returned results is more than 0, then the username is considered valid. The accuracy can be further improved by using profile patterns (see Table 1) and hand-tuning.

Evaluation Results

In order to analyze the competitiveness of the designed method, we performed a complete analysis on different communities. Twelve different well known communities were selected. For each community, a set of *username-username* pairs was selected, for which the base username was in the BlogCatalog community and the target one was in the community. The proposed method was employed in order to extract the set of possible usernames in the target community. The inclusion of the target username in this set, which on average has cardinality less than 5, is checked and the overall accuracy was recorded. The results showed that if the base username is from the BlogCatalog community, on average, our method has 63% accuracy, and in the best case, can be up to 78% accurate. As already mentioned, the base username was selected from the BlogCatalog community. We also decided to perform the same experiment with the base usernames from different communities. This allows us to analyze the accuracy variations depending on the base community. Tables 2 presents the detailed accuracy results when different base communities (rows) were used. On average, our method predicted the correct target-username in more than 66% of the cases and is up to 92% accurate in the best case scenario. Note that as highlighted in Table 2 certain communities have the tendency to be more useful in predicting the target username.

Conclusion and Future Work

In this paper, we have empirically studied the possibility of identifying corresponding identities across various communities on the web. Based on these evaluations, it turns out that usernames can be used quite successfully to identify corresponding usernames in various communities. We have also proposed a method to identify corresponding usernames in various communities. The method has been successfully evaluated over 12 different communities and thousands of usernames with the average accuracy of around 66%. In our future work, we aim to deal with the many challenges that we faced during the course of this research. For instance, there are many cases where same usernames does not necessarily guarantee the same identity. For example, while a username such as *hrlz1988prague* might represent the same identity, but common usernames such as *john.smith* can be employed by different identities in various communities and do not necessarily represent a unique individual.

Acknowledgements

This work is, in part, sponsored by AFOSR Grant FA95500810132.

References

Flake, G.; Lawrence, S.; and Giles, C. 2000. Efficient identification of Web communities. In *ACM SIGKDD*, 150–160. ACM, USA.

Hu, M., and Liu, B. 2004. Mining and summarizing customer reviews. In *ACM SIGKDD*, 168–177. ACM New York, NY, USA.

Wasserman, S., and Faust, K. 1994. *Social Network Analysis: Methods and Applications*. Cambridge University Press.