User Identification Across Social Media

REZA ZAFARANI*, Syracuse University
LEI TANG, Clari
HUAN LIU, Arizona State University

People use various social media sites for different purposes. The information on each site is often partial. When sources of complementary information are integrated, a better profile of a user can be built. This profile can help improve online services such as advertising across sites. To integrate these sources of information, it is necessary to identify individuals across social media sites. This paper aims to address the cross-media user identification problem. We provide evidence on the existence of a mapping among identities of individuals across social media sites, study the feasibility of finding this mapping, and illustrate and develop means for finding this mapping. Our studies show that effective approaches that exploit information redundancies due to users' unique behavioral patterns can be utilized to find such a mapping. This study paves the way for analysis and mining across social networking sites, and facilitates the creation of novel online services across sites. In particular, recommending friends and advertising across networks, analyzing information diffusion across sites, and studying specific user behavior such as user migration across sites in social media are one of the many areas that can benefit from the results of this study.

Categories and Subject Descriptors: H.2.8 [**Database Management**]: Database Applications—*Data mining*

General Terms: Theory, Measurement

Additional Key Words and Phrases: Social network analysis, user identification, cross-media analysis, behavioral modeling

ACM Reference Format:
Reza Zafarani, Lei Tang, and Huan Liu. 2015. User identification across social media. ACM Trans. Knowl. Discov. Data 10, 2, Article 16 (October 2015), 30 pages.
DOI: http://dx.doi.org/10.1145/2747880

1. INTRODUCTION

Advertisement revenue is often a principal sources of finance for a sustainable social networking site. Web giants such as Google report a $50.57 billion dollar yearly ad revenue[1]; that is 91% of Google's annual revenue[2]. The same consistent pattern is observed among other internet sites such as Facebook or Yahoo!. Thus, internet sites are often interested in increasing the success rate of their ad campaigns.

[1] http://bit.ly/1fbM89P.
[2] http://bit.ly/1k5uVXI.

*The majority of this work was conducted when Reza Zafarani was affiliated with Computer Science and Engineering at Arizona State University.

This work is supported in part by ONR grants N000141010091 and N000141410095.

Authors' addresses: R. Zafarani, Department of EECS, Syracuse University; L. Tang, Clari, Mountain View, California; H. Liu, Computer Science and Engineering, Arizona State University.

DOI: http://dx.doi.org/10.1145/2747880

It is well-known that the relevance of ads to the interests of individual users can directly impact the success of an ad campaign. To have relevant ads, it is required to have a good understanding of individuals, which can be achieved by profiling users. Though a growing number of people use social media, people use various social media for different purposes, and the information about an individual on each site is often limited. Though each site has only limited information about a user, other social media sites could provide complementary information for the user, and integrating information from various sites can help build better user profiles. However, for combining these sources of complementary information, one has to reliably identify corresponding user identities across social media sites. Companies such as Yahoo! often sign agreements with other companies to connect their user base for better marketing and a richer user experience. However, preliminary attempts to match users across sites even for these companies are challenging as users provide limited or no information for matching purposes [Communication 2013].

This paper proposes an alternative solution to connecting users across social media sites by exploiting the nature of social media and its networks. Connecting user identities across social media sites is not a straightforward task. The primary obstacle is that connectivity among user identities across different sites is often unavailable. This disconnection happens since most sites maintain the anonymity of users by allowing them to freely select usernames instead of their real identities, and also because different websites employ different user-naming and authentication systems. Moreover, websites rarely link their user accounts with other sites or adopt Single Sign-On technologies such as openID, where users can logon to different sites using a single username (e.g., users can login to Google+ and YouTube with their GMail accounts). Regardless, there exists a mapping between usernames across different sites that connects the real identities behind them. *Can we find this mapping?*

In this article, we provide evidence on the existence of a mapping among identities across multiple social media, study the feasibility of finding this mapping, and illustrate and develop means for finding it.

The need for identifying corresponding users across different social media is multifold. In addition to the aforementioned marketing example, we illustrate the need using multiple examples.

(1) **User Migrations.** Consider the migration of users in social media [Kumar et al. 2011]. Users often migrate from one social network to another due to their limited time and the better quality of service they receive at the destination network. Given a mapping among identities of users across these two networks and their membership dates (or dates where they started their activity on the destination network), a migration can be detected. The network from which users are migrating can decrease the migration rate by detecting it early and can also improve its site by introducing the additional features and services that the destination network provides.

(2) **Enhancing Friend Recommendation.** Better friend recommendations can help increase user engagement in social media sites. Often, nonconnected users that share mutual friends are recommended as potential friends. Consider the following example. John and Catherine are not connected and are both friends of Russ on social network S_1. Thus, Catherine seems a good candidate for recommendation to John on S_1. Catherine and John are also members of social network S_2 and are also not connected on S_2. Assume that Catherine and John share no mutual friends on S_2. With the information that we have from S_1, the recommendation algorithm could recommend Catherine to John on S_2, even though they share no mutual friends on S_2.

This type of recommendation is only possible when there is cross-site complementary information. Cross-site friendship information will increase the recall of the friendship recommendation algorithm by recommending more known friends, as well as its accuracy by having more information about the network.

(3) **Information Diffusion.** Information diffusion is commonly measured within the context of a single social network. In reality, information can flow within and across different social networks. Thus, it is of interest to investigate whether information diffuses more within one network or across networks. Moreover, what type of information propagates more within a network and what type propagates more across networks?

(4) **Multiple Network Group Interaction.** By connecting users across sites, one can analyze group interaction across sites. Multiple-network group interactions can be viewed as an instance of single-net group interactions by combining the graph of all connected social networks. Thus, methods proposed for single network group interaction analysis [Tang et al. 2012a] can be generalized for multiple networks.

(5) **Analyzing Network Dynamics.** Dynamics of single-site social networks are well-studied in the literature. These networks are known to have a power-law degree distribution, a small average path length, and being highly clusterable [Zafarani et al. 2014]. However, users belong to multiple sites and these network properties need to be generalized to multiple networks. In particular, it is interesting to determine how close the dynamics of single networks are to that of multi-networks.

To approach the earlier mentioned problems, one has to first identify users across sites. Our methodology for identifying users across sites is based on unique behavioral patterns that individuals exhibit on social media. Our methodology has direct roots in behavioral theories in sociology and psychology. These behaviors are due the environment, personality, or even human limitations of the individuals and are manifested in the content and link individuals generate on social media. Our methodology performs feature discovery [Scott and Matwin 1999; Cormack et al. 2007] to capture traces that these behaviors leave in social media for user identification. Before introducing our methodology, we discuss the types of information that can help us identify users across sites.

Network structure and friendship information is known to carry information that could prove useful in many tasks, such as link and attribute prediction, spam detection, behavioral analysis, and group behavior. Recent studies have indicated that link-based methods outperform many other techniques on various tasks. In Agrawal et al. [2003], the authors show that their link-based algorithm exhibits a significant accuracy advantage over the classical text-based methods for mining certain newsgroups. Moreover, it is well established that link-based methods are more resilient to spam attacks [Gyongyi et al. 2004]. Examples from social networks include systems that are designed using link-based methods that combat unwanted communications [Mislove et al. 2008] or that guard against Sybil attacks [Tran et al. 2009; Yu et al. 2006].

Recent interest in the information that immediate links (friends) carry about an individual has brought with it interesting results. When tracking link formation in online sites, Kossinets and Watts [2006], and on a larger scale Leskovec et al. [2008], found that the likelihood of forming links increases steadily as the number of common friends increases. In similar membership closure studies [Crandall et al. 2008], it has been shown that the same increasing trend can be observed when analyzing the probability of joining a community as a function of the number of friends who have already joined. In another study, Backstrom et al. [2006] show that the tendency of an individual to join a group is influenced not only by the number of friends the individual has within the community, but also crucially by how they are linked to one another.

These results suggest that it should be reasonable to use link information to identify users across social networks. The link information and in particular friends (immediate links) of an individual can form a "social signature" of the user that can be employed to identify the individual across networks. We will next detail the link-based approach taken in order to identify individuals across social networks in Section 2. This section paves the way for Section 3, which completes the description for our user identification methodology and is followed by a review of the related work in Section 4. We conclude and discuss our future work in Section 5.

2. LINK-BASED USER IDENTIFICATION

Let us formally define the problem of identifying individuals across social media sites. Without loss of generality, we focus on two social media sites and a single individual in this study. This is reasonable because solving the problem of two sites can be easily generalized to the problem of n sites by considering n sites in a pairwise manner. The same argument holds for more than one individual. Following the tradition in machine learning and data mining research, we solve the problem given some available *labeled information*. This labeled information is the *known* part of a one-to-one relationship that connects users that coexist on both networks. In this article, we call this labeled information "the mapping". The mapping for these two social networks contains a set of known individuals and their identities on both these networks; it basically denotes "who on this network is who on the other?". Finally, we focus on situations where the identity of the individual on one of these websites is known, for example, profile of someone is known on Twitter; can we find his profile on Facebook?

When using link information, a social network $\mathcal{S}$ is represented using a graph $G_{\mathcal{S}}(V_{\mathcal{S}}, E_{\mathcal{S}})$ and the identity of an individual is represented using a node v (vertex) in this social graph, that is, $v \in V_{\mathcal{S}}$. The mapping connects a node in the base-site's graph to its corresponding node in the target-site's graph.

Definition (*Link-Based User Identification*). Given two social media sites $\mathcal{S}_1$ (base-site) and $\mathcal{S}_2$ (target-site) and their respective social network graphs $G_{\mathcal{S}_1}(V_{\mathcal{S}_1}, E_{\mathcal{S}_1})$ and $G_{\mathcal{S}_2}(V_{\mathcal{S}_2}, E_{\mathcal{S}_2})$, a mapping $\mathcal{M} \subseteq V_{\mathcal{S}_1} \times V_{\mathcal{S}_2}$ that identifies a subset of users across these networks and an individual u whose identity (a vertex $v_i \in V_{\mathcal{S}_1}$) we know on $\mathcal{S}_1$ (base-node), a link-based user identification procedure attempts to resolve the identity (a vertex $v_j \in V_{\mathcal{S}_2}$) of u on S_2 (target-node).

We introduce two techniques to identify users across sites based on link information. The first technique uses only local information (i.e., neighborhoods and shared friends) to identify users across sites. The second techniques utilizes global network information (i.e., the whole graph) to identify users across sites.

2.1. A Local Link-Based Method for Identifying Users

We introduce an iterative method for identifying users across sites using local link information. The method considers users across sites that share most mutual friends across sites as identities of the same individual. Our intuition is that as users join multiple sites, it is more likely for them to become friends with individuals that they have befriended on other sites. So, nodes that share most common friends across sites are more likely to be the same user. Inspired by the success of methods that utilize common friends within one site, our method employs the same heuristic across sites. The method's pseudocode is outlined in Algorithm 1, in which, $\mathcal{F}(i, \mathcal{S})$ denotes friends of user i on site $\mathcal{S}$.

The method starts from the users not in the mapping, and it acts similar to the semisupervised learning algorithms and in particular co-training [Zhu 2005]. In the

ALGORITHM 1: The Link-based Iterative Method for Identifying Individuals

```
Input: $G_{\mathcal{S}_1}(V_{\mathcal{S}_1}, E_{\mathcal{S}_1})$, $G_{\mathcal{S}_2}(V_{\mathcal{S}_2}, E_{\mathcal{S}_2})$, Mapping $\mathcal{M}$, $v_1 \in V_{\mathcal{S}_1}$ (base-node)
Output: $v_2 \in V_{\mathcal{S}_2}$ (target-node) or NIL
shouldContinue = True, targetNode = NIL;
while shouldContinue do
    $\mathcal{M}_1 = \{i|(i, j) \in \mathcal{M}\}$, $\mathcal{M}_2 = \{j|(i, j) \in \mathcal{M}\}$; % Nodes in the Mapping
    if $V_{\mathcal{S}_1} \setminus \mathcal{M}_1 = \emptyset$ or $V_{\mathcal{S}_2} \setminus \mathcal{M}_2 = \emptyset$ then
        shouldContinue = False, break while; % No More Users Left
    end
    % Find Users with the Maximum Number of Friends among Mapping Nodes
    $x = \arg\max_i |\mathcal{F}(i, \mathcal{S}_1) \cap \mathcal{M}_1|$, s.t., $i \in V_{\mathcal{S}_1} \setminus \mathcal{M}_1$;
    $y = \arg\max_j |\mathcal{F}(j, \mathcal{S}_2) \cap \mathcal{M}_2|$, s.t., $j \in V_{\mathcal{S}_2} \setminus \mathcal{M}_2$;
    if $x = v_1$ then
        targetNode = y, shouldContinue = False, break while; % Target Found
    end
    $\mathcal{M} = \mathcal{M} \cup \{(x, y)\}$; % Add an Identified Pair to the Mapping
end
Return targetNode;
```

pseudocode, the users already mapped in S_1 (S_2) are denoted as $\mathcal{M}_1$ ($\mathcal{M}_2$), and the users not mapped are denoted as $V_{\mathcal{S}_1} \setminus \mathcal{M}_1$ ($V_{\mathcal{S}_2} \setminus \mathcal{M}_2$).

The method then maps two users to one another across networks based on their number of friends inside the mapping. Here, we find two users, one on each network, who have the most number of friends among users in the mapping, and we assume these users represent the same individual.

Since these two users are assumed to represent the same individual, they are added to the mapping.

This process is continued until no further user is identified on both networks ($V_{\mathcal{S}_1} \setminus \mathcal{M}_1 = \emptyset$ or $V_{\mathcal{S}_2} \setminus \mathcal{M}_2 = \emptyset$), or the required user is found on both networks.

The method only considers the local neighborhood of nodes. Our next method considers global network structure to identify users across sites.

2.2. A Global Link-Based Method for Identifying Users

The local algorithm only considers nodes in the mapping that are one-hop away. The algorithm can be modified in order to consider nodes in the mapping that are more than one-hop away. For each node, the number of nodes in the mapping that are 1...k hops away can be computed and a k-dimensional vector can be used to represent users. The distance between these vectors could help identify the identities of the same individual and in turn, grow the mapping. A more sophisticated approach is to use the topology of the induced subgraphs of the nodes in the mapping and the nodes connected to them. We can assume that the two networks are two different views of the same underlying structure. In other words, we assume that users possess a specific friendship behavior and the way they befriend others across different networks are just different ways that they exhibit this behavior. We expect these networks to be highly correlated and thus a transformation between them can be computed.

The base-site and target-site graph can be represented as an adjacency matrix. Let us call these matrices A_1 and A_2. An additional preprocessing step is usually taken in order to extract structural features of the graphs. For preprocessing purposes, the normalized Laplacians, $\mathcal{L}_1$ and $\mathcal{L}_2$, for each graph is calculated. The normalized Laplacian $\mathcal{L}$ for adjacency matrix A is calculated as follows

$$\mathcal{L} = D^{-1/2} L D^{-1/2}, \tag{1}$$

$$L = D - A, \tag{2}$$

where D, also known as the degree matrix, is a diagonal matrix where each entree on the diagonal represents the degree of the node. L here represents the unnormalized Laplacian matrix. After computing the normalized Laplacians, the k top eigenvectors of the matrix are extracted and are used instead of the adjacency matrix. This matrix can better represent the structural features of the graph when compared with the adjacency matrix [Tang et al. 2012c]. Different k's were tested in our experiments, $k = 3, 5, 20, 50, 100, 200, 500$. For values above 50, our results did not improve much; therefore, we used $k = 100$ for our experiments. Let us call these new matrices X_1 and X_2. We take the mapping part of these two matrices (corresponding mapped rows) and call them X_1^m and X_2^m. Assuming there exists a linear transformation, the transformation W can be found using the following optimization

$$\min \left|\left|X_1^m W - X_2^m\right|\right|_2. \tag{3}$$

The transformation W can be efficiently computed using a least square approximation. After the weights are obtained, the unmapped part of matrix X_1 can be multiplied by W and then compared with the unmapped part of X_2. Rows (users) with the highest similarity are assumed to be the same individual.

2.3. Empirical Study

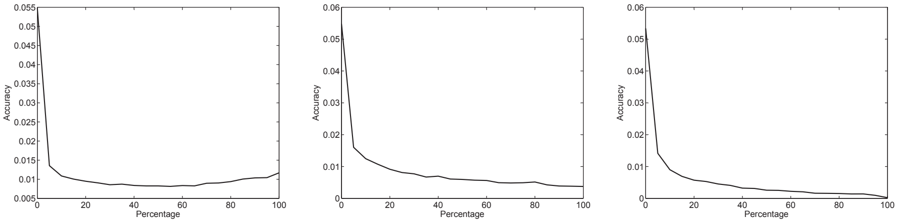
2.3.1. Evaluating the Link-Based Method.

Evaluation with Synthetic Data. To conduct a systematic evaluation of the proposed methods, we generated a set of synthetic datasets. These synthetic datasets must contain mapping information (labeled data). For synthetic dataset generation, we adhere to the following procedure: (1) a real-world social network was gathered and used as the base-site; (2) the base-site's network was copied as the target-site; and (3) noise was introduced on the target-site. Three common types of noise were employed, namely: (i) randomly adding edges to the target-site with probability p, (ii) randomly removing edges from the target-site with probability p, or (iii) randomly rewiring [Watts and Strogatz 1998] edges from the target-site with probability p, $0\% \leq p \leq 100\%$. In rewiring, for every disjoint pair of random edges (a, b), (c, d), we swap their end points to get new edges, (a, c), (b, d). This makes sure that the degrees are preserved for every node in the target-site graph. Based on the types of noise introduced and the probability value p, we call these datasets $SYN\_ADD(p)$, $SYN\_REMOVE(p)$, and $SYN\_REWIRE(p)$, respectively. The mapping is obvious in the case of synthetic data, and for every node in the base-site, the mapping connects it to the corresponding copied node in the target-site. For the real-world network used in our synthetic dataset generation, we employed a collection of 11 large scale social media datasets (see Table I) obtained from the social computing data repository [Zafarani and Liu 2009b].

We conduct experiments on synthetic data to verify if our link-based methods perform effectively in a controlled environment. We start with no noise ($p = 0$) and notice that the local method is not even accurate for cases where no noise is introduced. This is a result of many nodes having the same number of friends among mapping nodes. Table I shows the accuracy rate of both methods in the case where no noise is introduced over all synthetic datasets. The table shows that the local method is, in eight out of eleven cases, less than 2% accurate, and the best accuracy rate obtained is less than 7%. On the contrary, the global model is highly accurate with no noise and is at least 79% accurate and at times, up to 98% accurate. Next, we added noise. We used BlogCatalog dataset as the real network required for synthetic data generation. Part of the mapping was used for training and the rest for testing. 10-fold cross-validation was used and the average accuracy for correctly predicting identities in the testing part of the mapping was recorded. Figure 1 depicts these accuracy rates for the local method and for cases

Table I. Prediction Accuracy for Different Social Networks

Site	Nodes (Mapping size)	Edges (Friendship links)	Accuracy (Local method)	Accuracy (Global method)
Blogcatalog	88,784	4,186,390	6.93%	89.3%
Buzznet	101,168	4,284,534	5.11%	79.7%
Digg	116,893	7,261,524	1.81%	91.4%
Douban	154,907	654,188	1.78%	84.1%
Flixster	2,523,386	9,197,338	0.57%	96.6%
Friendster	100,199	14,067,887	0.32%	91.3%
Foursquare	106,218	3,473,834	0.53%	98.0%
Hyves	1,402,611	2,777,419	0.37%	95.0%
Last.fm	108,493	5,115,300	0.76%	95.6%
Livemocha	104,438	2,196,188	4.57%	96.4%
YouTube	1,138,499	2,990,443	4.57%	90.5%



(a) Edge Addition (b) Edge Removal (c) Edge Rewiring

Fig. 1. Prediction accuracy for different percentage of edges added/removed/rewired.

where with different probabilities, edges were being added, removed, or rewired. As seen in these figures, the local method performs quite poorly on synthetic data. The average accuracy rates for *SYN_ADD*(p), *SYN_REMOVE*(p), and *SYN_REWIRE*(p) were 4%, 1%, and 1%, respectively. The results did not improve much for the global method. With $p = 0.5$, the accuracy rates for *SYN_ADD*(p), *SYN_REMOVE*(p), and *SYN_REWIRE*(p) were 6%, 10%, and 0.01%, respectively. Next, we evaluate the performance of the methods with real-world datasets.

Evaluation with Real-World Data. We gathered two real-world datasets. For collecting real-world datasets, we require additional mapping information about identities across social media sites. Fortunately, there exist websites where users have the opportunity of listing their identities (user accounts) on different social networks. For instance, on Facebook users can list their usernames on different sites. This can be thought of as *labeled* data for our learning task since it provides the accurate mapping for our experiments. In addition to labeled data, these websites provide strong evidence on the existence of a mapping between identities across social media sites. Later on, in Section 3.2.1, we discuss the procedure for collecting mapping information in detail. From sites that provide such mapping information, we gathered individuals that had account on two sites: Flickr and BlogCatalog, due to their large network size and many overlaps.

We collected two disjoint sets of individuals. All individuals had accounts on both BlogCatalog and Flickr. We call these sets $\mathcal{I}_1$ and $\mathcal{I}_2$ ($\mathcal{I}_2 \cap \mathcal{I}_2 = \emptyset$). For each member of these sets, we collected their identity on both BlogCatalog and Flickr. Then for individuals in $\mathcal{I}_1$ and for each of their two identities, we collected all the users who were within a three-hop distance in the respective network using a Breadth-First-Search crawling procedure [Menczer 2007]. For $\mathcal{I}_2$, however, we only crawled users

Table II. Real-World Dataset Properties

Dataset	BlogCatalog network size	Flickr network size	Mapping size $\lvert\mathcal{M}\rvert$
$BF3Hop$	88,784 users	564,491 users	1,747 individuals $\lvert\mathcal{I}_1\rvert$
$BF1Hop$	1,455 users	630 users	546 individuals $\lvert\mathcal{I}_2\rvert$

Table III. Performance of Link-Based Methods on Real-World Datasets

Dataset	Local method	Global model
$BF3Hop$	≈0	≈0
$BF1Hop$	0.3%	0.6%

who were within a one-hop distance (immediate friends). Hereafter, we will refer to the network datasets created from $\mathcal{I}_1$ and $\mathcal{I}_2$ as $BF3Hop$ and $BF1Hop$, respectively. Table II provides some statistics about the cardinalities of these datasets.

These datasets help showcase the effect of nonimmediate link information on the performance of our proposed algorithms. This is true since $BF3Hop$ contains nonimmediate information, whereas $BF1Hop$ lacks this property.

We evaluate both methods on real-world datasets. We apply the local method to our real-world datasets and 10-fold cross-validation is employed to measure accuracy. The method failed on both datasets with an average accuracy rate of 0.3% on $BF1Hop$ and ≈ 0 on $BF3Hop$. Similarly, we evaluated the global model. However, the results did not improve much. For real-world datasets, the accuracy rate were 0.6% on $BF1Hop$ and 0% on $BF3Hop$. Table III summarizes the results of link-based methods on the real-world datasets.

We have shown that using both local and global information, poor performances are expected when using real-world datasets. The question is whether *there are any properties in real-world datasets that need to be considered in order to obtain higher accuracy rates*. We investigate this question next.

2.3.2. Investigating Properties of Real-World Datasets. To further investigate this, let us present various hypotheses regarding the properties of the users that are in the mapping. These link-related properties that identities share when representing the same individual across different networks can be employed when designing methods for identifying users across social networks. Each of these hypotheses is empirically evaluated. The observations gathered while evaluating these hypotheses can be used later to help construct link-based methods.

To simplify the notation in these hypotheses, let x_i be a user (node), and $\mathcal{F}(x_i, \mathcal{S})$ the set of friends user x_i has on site $\mathcal{S}$. For two users $x_1 \in \mathcal{S}_1$, $x_2 \in \mathcal{S}_2$ that belong to two different sites, we define the concept of *shared-friends across networks*. In this case, they are the set of people who coexist on both $\mathcal{S}_1$ and $\mathcal{S}_2$ and are friends with both x_1 and x_2. For clarity, shared friends are depicted in Figure 2. In this figure, the mapping consists of three pairs and is shown using dashed lines and black circles denote shared friends between x and y. The concept is formalized as follows

$$\begin{aligned}\mathcal{SF}(x_1, x_2) \;=\; \{(y_1, y_2) \mid y_1 \in \mathcal{S}_1, y_2 \in \mathcal{S}_2, (y_1, y_2) \in \mathcal{M}, \\ y_1 \in \mathcal{F}(x_1, \mathcal{S}_1), y_2 \in \mathcal{F}(x_2, \mathcal{S}_2)\}.\end{aligned}$$

We also define the concept of *crossed-over friends* for a user x. These are the corresponding identities, on the *other* site, for the friends of x who are members of both sites. So, if x is a member of $\mathcal{S}_1$, this set includes identities on S_2 for those friends of x that are members of both sites. Formally

$$\mathcal{CR}_{\mathcal{S}_1 \to \mathcal{S}_2}(x) = \{y \mid y \in \mathcal{S}_2, \exists x' \in \mathcal{F}(x, \mathcal{S}_1), s.t., (x', y) \in \mathcal{M}\}.$$

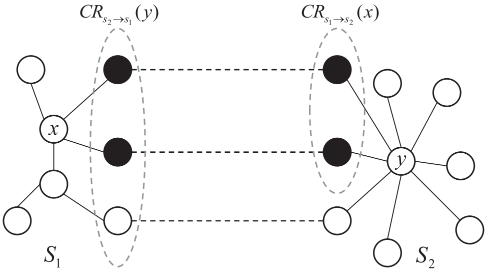


Fig. 2. A visualization of two social networks and the mapping. Social network S_1 consists of the nodes on the left and social network S_2 consists of the nodes on the right. Dashed lines denote the mapping $\mathcal{M}$ ($|\mathcal{M}| = 3$), solid circles denote shared friends $\mathcal{SF}(x, y)$, circles in the right dashed oval denote crossed-over friends $\mathcal{CR}_{\mathcal{S}_1 \to \mathcal{S}_2}(x)$, and circles in the left dashed oval denote $\mathcal{CR}_{\mathcal{S}_2 \to \mathcal{S}_1}(y)$.

Table IV. Friends Shared Across Social Networks

Property	*BF*1*Hop*	*BF*3*Hop*
Average number of friends shared	1.14	.18
Average number of friends on Flickr	3.08	26.22
Average number of friends on BlogCatalog	24.89	141.41
Average % Flickr friends shared	37%	2%
Average % BlogCatalog friends shared	9%	.2%
Maximum number of friends shared	32	30
Minimum number of friends shared	0	0
Standard deviation of the number of friends shared	2.30	1.09

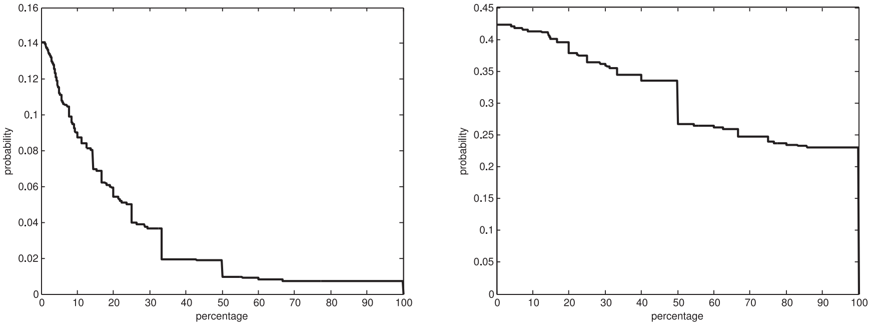
This definition is bidirectional. Note that if users x_1 and x_1' belong to the same individual, that is, $(x_1, x_1') \in \mathcal{M}$, then the value of $|\mathcal{CR}_{\mathcal{S}_1 \to \mathcal{S}_2}(x)|$ is *not* necessarily equal to $|\mathcal{CR}_{\mathcal{S}_2 \to \mathcal{S}_1}(x')|$. In general, for any two users $x \in \mathcal{S}_1$ and $y \in \mathcal{S}_2$ there could be no relationships between the values of $|\mathcal{CR}_{\mathcal{S}_1 \to \mathcal{S}_2}(x)|$, $|\mathcal{CR}_{\mathcal{S}_2 \to \mathcal{S}_1}(y)|$, and $|\mathcal{SF}(x, y)|$, for example, consider the situation where there are no shared friends but different number of crossed-over friends. Similarly, in Figure 2, circles in the right dashed oval denote $\mathcal{CR}_{\mathcal{S}_1 \to \mathcal{S}_2}(x)$, and circles in the left dashed oval represent $\mathcal{CR}_{\mathcal{S}_2 \to \mathcal{S}_1}(y)$. Given these formal definitions, we present our hypothesis next.

2.4. Hypotheses Verification

$\mathcal{H}_1$: **There is a correlation between the number of friends of the same individual across networks.** To test this, for all the users in the mapping, we analyze the number of friends they have in both networks. A Pearson correlation analysis revealed that the number of friends are uncorrelated across networks for the same individual. The correlation coefficient ρ was 0.038 for *BF*3*Hop* and 0.186 for *BF*1*Hop*. For a randomly generated mapping, the correlation coefficient ρ was 0.007 for *BF*3*Hop* and 0.019 for *BF*1*Hop*. This shows that there is no strong correlation among the number of friends across networks for the same individual.

$\mathcal{H}_2$: **There is a correlation between the percentage of friends of the same individual on each network that are shared across networks.** To verify this, we first enumerated the number of friends shared between identities of the same individual across networks, that is, we calculated $\mathcal{SF}(x_1, x_2)$ for all $(x_1, x_2) \in \mathcal{M}$, and for both datasets. Table IV shows some statistics about these shared friends.

As shown in this table, the average number of friends shared is at most around 1 in the datasets. Having at most one shared friend suggests that the friends that are shared across both social networks, in the best case, can form connected components



(a) Flickr to BlogCatalog (b) BlogCatalog to Flickr

Fig. 3. Target user connection probability to different fractions of crossed-over friends.

on both networks. Starting from an individual in the mapping and its two identities, a Breadth-First-Search procedure on each network should be able to traverse many other users in the mapping.

The table also shows, for both BlogCatalog and Flickr, the average values for the percentage of users' friends that were shared. The small values of these percentages denotes that many friends on both social networks do not cross over into the other[3]. A correlation analysis on these percentages across networks, when there was at least one friend shared, showed that $\rho \approx 0$ for both datasets. Again, the value was close to the correlation coefficient for both datasets when the mapping was randomly generated and shows that there is no strong correlation between percentages.

$\mathcal{H}_3$: **The target-node is connected to the crossed-over friends of the base-node.** Here, we conjectured intuitively and based on previous evidence from the social sciences (e.g., see Herding Behavior [Easley and Kleinberg 2010]), that when users join various social networks, their friends also follow them and join these networks. We assume that if one analyzes the connections of crossed-over friends, one might be able to find the user on the target network.

For evaluating this hypothesis, for all pairs $(x_1, x_2) \in M$, $x_1 \in \mathcal{S}_1$, $x_2 \in \mathcal{S}_2$, we first extracted all crossed-over friends of x_1 ($\mathcal{CR}_{\mathcal{S}_1 \to \mathcal{S}_2}(x_1)$). Then for all members of this set $y \in \mathcal{CR}_{\mathcal{S}_1 \to \mathcal{S}_2}(x_1)$, we checked whether the target-node x_2 is connected to y, that is, $x_2 \in \mathcal{F}(y, \mathcal{S}_2)$. In other words, we are trying to calculate the probability of identifying the target-node by analyzing the connections of the crossed-over friends of the base-node.

It turns out that in both datasets, the probability of target-node x_2 being connected to *all* the friends of the base-node that crossed-over is always less than 5%. Furthermore, the probability of x_2 being connected to *at least one* of the friends is still very low for both datasets (around 45% for $BF3Hop$ dataset at its best). Figure 3 shows the probability of the target-node being connected to different fractions of crossed-over friends of the base-node for the $BF3Hop$ dataset: (a) friends crossed-over from Flickr to BlogCatalog, and (b) from BlogCatalog to Flickr. For instance, Figure 3(b) shows that in the best case, one has less than a 45% chance to find the target-node based on crossed-over friends of the base-node. This is because in 55% of the cases, the target-user is not even connected to these friends. The 45% is reduced to less than 5% in the worst case. But,

[3] This could also be due to the small size of the mapping in the dataset; however, when collecting the initial set of mapping users from BlogCatalog we made sure a connected component was collected to reduce the effect of this phenomenon.

when the user is connected to these friends, is it easy to distinguish him from others who are also connected to these friends? This brings us to our next hypothesis.

$\mathcal{H}_4$: **If the target-node is connected to the crossed-over friends of the base-node, how easily can it be identified?** To answer this question, we further analyzed these crossed-over friends and ranked other users in the target network based on the number of connections they have to them. In these ranked users, we found that in $BF1Hop$ and on average, the target user x_2's ranking is 19 for friends who cross-over from BlogCatalog to Flickr and 25 in the opposite direction. These averages showed a dramatic increase in $BF3Hop$ and were 272 and 251, respectively. Furthermore, in $BF1Hop$, x_2 was the top ranked user in only 23% of the cases where friends crossed over from BlogCatalog to Flickr and 24% of the cases where the crossing over took place in the opposite direction. These percentages dropped to 9% and 8% for the $BF3Hop$ dataset, respectively. Note that even if one is successful in finding that the target user among the nodes that are connected to the crossed over friends of the base-node, it is very unlikely to correctly identify the target user. For example, in case of friends who crossed over from BlogCatalog to Flickr in $BF3Hop$, this probability is at most $45\% \times 9\% \approxeq 4\%$.

The results from the hypotheses verification suggest that methods that deal with link information can perform poorly when solving the user identification problem. Based on the evidence that we gathered, it is very unlikely to come up with new methods that can perform better than the present methods if only link-information is employed. While our results clearly show that link information is not always useful, there could be cases where link information can be utilized for user identification across sites. This has been witnessed in recent studies where link-information has been successfully utilized to identify individuals across sites [Tang et al. 2012b; Liu et al. 2013; Zhang et al. 2014].

In summary, our results show that counter-intuitively, link information is not sufficient for identifying individuals across networks. In addition, link information might not be always available across sites for a general solution to the problem of user identification across sites. Therefore, we consider using content information to identify individuals.

3. BEYOND LINK INFORMATION

To use content information to identify users across social networks, we introduce a methodology (MOBIUS) [Zafarani and Liu 2013] for finding the mapping among identities across social media sites. Our methodology is based on behavioral patterns that users exhibit in social media, and has roots in behavioral theories in sociology and psychology. Unique behaviors due to environment, personality, or even human limitations can create redundant information across social media sites. Our methodology exploits such redundancies for identifying users across social media sites. We use the minimum amount of content information available across sites and discuss how additional information can be added.

Let us begin by formulating our problem in terms of content information. Information shared by users on social media sites provides a *social fingerprint* of them and can help identify users across different sites. We start with the *minimum* amount of information that is available on *all* sites. Later on, in Section 3.3, we will discuss how one can add extra information to this minimum as it becomes available across sites. In terms of information availability, *usernames* seem to be the minimum common factor available on all social media sites. Usernames are often alphanumeric strings or email addresses, without which users are incapable of joining sites. Usernames are unique on each site and can help identify individuals, whereas most personal information, even "first name + last name" combination, are nonunique. We formalize our problem using usernames as the atomic entities available across all sites. Other profile attributes, such as gender, location, interests, profile pictures, language, and so on, when added to usernames, should help better identify individuals; however, the lack of consistency in the available information across all social media, directs us toward formulating with usernames. When considering usernames, two general problems need to be solved for user identification:

I. Given two usernames u_1 and u_2, can we determine if they belong to the same individual?
II. Given a single username u from individual $\mathcal{I}$, can we find other usernames of $\mathcal{I}$?

Question **II** can be answered via a two-stage process: (1) we find the set of all usernames C that are likely to belong to individual $\mathcal{I}$. We denote set C as *candidate usernames* and, (2) for all candidate usernames $c \in C$, we check if c and u belong to the same individual. Therefore, if candidate usernames C are known, question **II** reduces to question **I**. Now, where can we find these candidate usernames?

We will discuss this later in our discussion section (Section 3.3) and from now on, we focus on question **I**. One can answer question **I** by learning an *identification function* $f(u, c)$,

$$f(u, c) = \begin{cases} 1 & \text{If } c \text{ and } u \text{ belong to same } \mathcal{I}; \\ 0 & \text{Otherwise.} \end{cases} \tag{4}$$

Without loss of generality, we can assume that username u is known to be owned by some individual $\mathcal{I}$ and c is the candidate username whose ownership by $\mathcal{I}$ we would like to verify. In other words, u is the prior information (history) provided for $\mathcal{I}$. Our function can be generalized by assuming that our prior is a *set* [4] of usernames $U = \{u_1, u_2, \ldots, u_n\}$ (hereafter referred to as "prior usernames"). Informally, the usernames of an individual on some sites are given and we have a candidate username on another site whose ownership we need to verify; for example, usernames u_t and u_f of someone are given on Twitter and Facebook, respectively; can we verify if c is her username on Flickr?

Definition (*Content-Based User Identification*). Given a set of n usernames (prior usernames) $U = \{u_1, u_2, \ldots, u_n\}$, owned by individual $\mathcal{I}$ and a candidate username c, a user identification procedure attempts to learn an identification function $f(.)$ such that

$$f(U, c) = \begin{cases} 1 & \text{If } c \text{ and set } U \text{ belong to } \mathcal{I}; \\ 0 & \text{Otherwise.} \end{cases} \tag{5}$$

Our methodology for ***MO****deling* ***B****ehavior for* ***I****dentifying* ***U****sers across* ***S****ites* (**MOBIUS**)[5] is outlined in Figure 4. When individuals select usernames, they exhibit certain behavioral patterns. This often leads to *information redundancy*, helping learn the identification function. In MOBIUS, these redundancies can be captured in terms of data features. Following the tradition in machine learning and data mining research, the identification function can be learned by employing a supervised learning framework that utilizes these features and prior information (*labeled data*), in our case, sets of usernames with known owners. Supervised learning in MOBIUS can be performed via either classification or regression. Depending on the learning framework, one can

[4] Mathematically, a set can only contain distinct values; however, here a user may use the same username on more than one site. In our definition of username set, it is implied that usernames are distinct when used on different sites, even though they can consist of the same character sequence.
[5] The resemblance to the Möbius strip comes from its *single-boundary* (representing a single individual) and its *connectedness* (representing connected identities of the individual across social media).

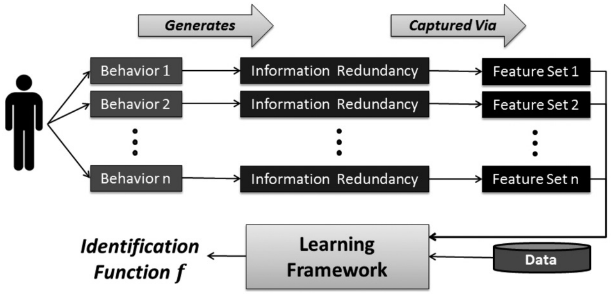


Fig. 4. MOBIUS: Modeling behavior for identifying users across sites.

even learn the probability that an individual owns the candidate username, generalizing our binary f function to a probabilistic model ($f(U, c) = p$). This probability can help select the most likely individual who owns the candidate username. The learning component of MOBIUS is the most straightforward. Thus, we next elaborate how to analyze behavioral patterns related to user identification and how features can be constructed to capture information redundancies due to these patterns. To summarize, MOBIUS contains (1) *behavioral patterns*, (2) *features* constructed to capture information redundancies due to these patterns, and (3) a *learning* framework. Given the interdependent nature of behaviors and feature construction, we discuss them together next.

3.1. MOBIUS: Behavioral Patterns and Feature Construction

Individuals often exhibit consistent behavioral patterns while selecting their usernames. These patterns result in information redundancies that help identify individuals across social media sites.

Individuals can avoid such redundancies by selecting usernames on different sites in a way such that they are completely different from their other usernames. In other words, their usernames are so different that given one username, no information can be extracted regarding the others. Theoretically, to achieve these independent usernames, one needs to select a username with Maximum Entropy [Cover and Thomas 2006]. That is, a **long** username string, as long as the site allows, with characters from those that the system permits, with **no redundancy** - an entirely **random** string.

Unfortunately, all of these requirements are contrary to human abilities [Yan et al. 2000]. Humans have difficulty storing long sequences with short-term memory capacity of 7 ± 2 items [Miller 1956]. Human memory also has limited capability in storing random content and often, selectively stores content that contains familiar items known as "chunks" [Miller 1956]. Finally, human memory thrives on redundancy, and humans can remember material that can be encoded in multiple ways [Paivio 1983]. These limitations result in individuals selecting usernames that are generally *not long*, *not random*, and have *abundant redundancy*. These properties can be captured using specific features which in turn can help learn an identification function. In this study, we find a set of consistent behavioral patterns among individuals while selecting usernames. These behavioral patterns can be categorized as follows:

(1) **Patterns due to human limitations.**
(2) **Exogenous factors.**
(3) **Endogenous factors.**

The features designed to capture information generated by these patterns can be divided into three categories:

(1) **(Candidate) Username Features**: these features are extracted directly from the candidate username c, for example, its length,
(2) **Prior-Usernames Features**: these features describe the set of prior usernames of an individual, for example, the number of observed prior usernames, and
(3) **Username↔Prior-Usernames Features**: these features describe the relation between the candidate username and prior usernames, for example, their similarity.

We will discuss behaviors in each of the earlier mentioned categories, and features that can be designed to harness the information hidden in usernames as a result of the pattern's existence. Note that these features may or may not help in learning an identification function. As long as these features could be obtained for learning the identification function, they are added to our feature set. Later on, in Section 3.2, we will analyze the effectiveness of all features, and if it is necessary to find as many features as possible.

3.1.1. Patterns Due to Human Limitations.

In general, as humans, we have (1) *limited time and memory* and (2) *limited knowledge*. Both create biases that can affect our username selection behavior.

(1) **Limitations in Time and Memory**

Selecting the Same Username. As studied recently [Zafarani and Liu 2009a], 59% of individuals prefer to use the same username(s) repeatedly, mostly for ease of remembering. Therefore, when a candidate username c is among prior usernames U, that is a strong indication that it may be owned by the same individual who also owns the prior usernames. As a result, we consider the number of times candidate username c is repeated in prior usernames as a feature.

Username Length Likelihood. Similarly, users commonly have a limited set of potential usernames from which they select one, once asked to create a new username. These usernames have different lengths and, as a result, a *length distribution* $\mathcal{L}$. Let l_c be the candidate username length and l_u be the length for username $u \in U$ (prior usernames). We believe that for any new username, it is more likely to have

$$\min_{u \in U} l_u \leq l_c \leq \max_{u \in U} l_u; \tag{6}$$

For example, if an individual is inclined to select usernames of length 8 or 9, it is unlikely for the individual to consider creating usernames with lengths longer or shorter than that. Therefore, we consider the candidate username's length l_c and the length distribution $\mathcal{L}$ for prior usernames as features. The length distribution can be compactly represented by a fixed number of features. We describe distribution $\mathcal{L}$, observed via discrete values $\{l_u\}_{u \in U}$ as a 5-tuple feature

$$\left(\mathbb{E}[l_u], \sigma[l_u], med[l_u], \min_{u \in U} l_u, \max_{u \in U} l_u\right), \tag{7}$$

where $\mathbb{E}$ is the mean, σ is the standard deviation, and med is the median of the values $\{l_u\}_{u \in U}$, respectively. Note that this procedure for compressing distributions as a fixed number of features can be employed for discrete distributions $\mathcal{D}$, observed via discrete values $\{d_i\}_{i=1}^{n}$.

Unique Username Creation Likelihood. Users often prefer not to create new usernames. One might be interested in the effort users are willing to put into creating new usernames. This can be approximated by the number of unique usernames

($uniq(U)$) among prior usernames U

$$uniqueness = \frac{|uniq(U)|}{|U|}. \tag{8}$$

Uniqueness is a feature in our feature set. One can think of $1/uniqueness$ as an individual's *username capacity*, that is, the average number of times an individual employs a username on different sites before deciding to create a new one.

(2) **Knowledge Limitation**

Limited Vocabulary. Our vocabulary is limited in any language. It is highly likely for native speakers of a language to know more words in that language than individuals speaking it as a second language. We assume the individual's vocabulary size in a language is a feature for identifying them, and as a result, we consider the number of dictionary words that are substrings of the username as a feature. Similar to *username length* feature, the number of dictionary words in the candidate username is a scalar; however, when counting dictionary words in prior usernames, the outcome is a distribution of numbers. We employ the technique outlined in Equation (7) for compressing distributions to represent this distribution as features.

Limited Alphabet. Unfortunately, it is a tedious task to consider dictionary words in all languages, and this feature can be used for a handful of languages. Fortunately, we observe that the alphabet letters used in the usernames are highly dependent on language. For instance, while letter x is common when a Chinese speaker selects a username, it is rarely used by an Arabic speaker, since no Arabic word transliterated in English contains letter x [Habash et al. 2007]. So, we consider the number of alphabet letters used as a feature, both for the candidate username as well as prior usernames.

3.1.2. Exogenous Factors. Exogenous factors are behaviors observed due to cultural affects or the environment that the user is living in.

Typing Patterns. One can think of keyboards as a general constraint imposed by the environment. It has been shown [Doctorow 2012] that the layout of the keyboard significantly impacts how random usernames are selected; for example, `qwer1234` and `aoeusnth` are two well-known passwords commonly selected by QWERTY and DVORAK users, respectively. Most people use one of two well-known keyboards DVORAK and QWERTY (or slight variants such as QWERTZ or AZERTY) [Wikipedia 2015]. To capture keyboard-related regularities, we construct the following 15 features for each keyboard layout (a total of 30 for both),

(1) (1 feature) The percentage of keys typed using the *same hand* used for the previous key. The higher this value the less users had to change hands for typing.
(2) (1 feature) Percentage of keys typed using the *same finger* used for the previous key.
(3) (8 features) The percentage of keys typed using each finger. Thumbs are not included.
(4) (4 features) The percentage of keys pressed on rows: Top Row, Home Row, Bottom Row, and Number Row. Space bar is not included.
(5) (1 feature) The approximate *distance* (in meters) traveled for typing a username. Normal typing keys are assumed to be (1.8cm)^2 (including gap between keys).

We construct these features for candidate username and each prior username. Thus, over all prior usernames, each feature has a set of values. Adopting the technique outlined in Equation (7) for compressing distributions as features, we construct $15 \times 5 = 75$ additional features for prior usernames.

Language Patterns. In addition to environmental factors, cultural priors such as language also affect the username selection procedure. Users often use the same or the same set of languages when selecting usernames. Therefore, when detecting languages of different usernames belonging to the same individual, one expects fairly consistent results. We consider the language of the username as a feature in our dataset. To detect the language, we trained an *n*-gram statistical language detector [Dunning 1994] over the European Parliament Proceedings Parallel Corpus[6], which consists of text in 21 European languages (*Bulgarian, Czech, Danish, German, Greek, English, Spanish, Estonian, Finnish, French, Hungarian, Italian, Lithuanian, Latvian, Dutch, Polish, Portuguese, Romanian, Slovak, Slovene,* and *Swedish*) from 1996–2006 with more than 40 million words per language. The trained model detects the candidate username language, which is a feature in our feature set. The language detector is also used on prior usernames, providing us with a language distribution for prior usernames, which again is compressed as features using Equation (7). The *detected language* feature is limited to European languages. Our language detector will not detect other languages. The language detector is also challenged when dealing with words that may not follow the statistical patterns of a language, such as location names, and so forth. However, these issues can be tackled from a different angle as we discuss next.

3.1.3. Endogenous Factors. Endogenous factors play a major role when individuals select usernames. Some of these factors are due to (1) personal attributes (name, age, gender, roles and positions, and so forth) and (2) characteristics, for example, a female selecting username `fungirl09`, a father selecting `geekdad`, or a PlayStation 3 fan selecting `PS3lover2009`. Others are due to (3) habits such as abbreviating usernames or adding prefixes/suffixes.

(1) **Personal Attributes and Personality Traits**

Personal Information. As mentioned, our language detection model is incapable of detecting several languages, as well as specific names, such as locations, or others that are of specific interest to the individual selecting the username. For instance, the language detection model is incapable of detecting the language of usernames `Kalambo`, a waterfall in Zambia, or `K2` and `Rakaposhi`, both mountains in Pakistan. However, the patterns in these words can be captured by analyzing the alphabet distribution. For instance, a user selecting username `Kalambo` most of the time will create an alphabet distribution where letter '*a*' is repeated twice more than other letters. Thus, we save the alphabet distribution of both candidate username and prior usernames as features. This will easily capture patterns like an excessive use of '*i*' in languages such as Arabic or Tajik [Ferguson 1957; Cowan 1958], where language detection fails. Another benefit of using alphabet distribution is that not only it is language-independent, but it can also capture words that are meaningful only to the user.

Username Randomness. As mentioned before, individuals who select totally random usernames generate no information redundancy. One can quantify the randomness of usernames of an individual and consider that as a feature that can describe individuals and help identify them. For measuring randomness, we consider the entropy [Cover and Thomas 2006] of the candidate username's alphabet distribution as a feature. We also measure entropy for each prior username. This results in an entropy distribution that is encoded as features using aforementioned technique in Equations (7).

[6] http://www.statmt.org/europarl/.

(2) **Habits**

"Old habits, die hard", and these habits have a significant effect on how usernames are created. Common habits are

Username Modification. Individuals often select new usernames by changing their previous usernames. Some

(a) add prefixes or suffixes
    —For example, `mark.brown` → `mark.brown2008`,
(b) abbreviate their usernames
    —For example, `ivan.sears` → `isears`, or
(c) change characters or add characters in between,
    —For example, `beth.smith` → `b3th.smith`.

Any combination of these operations is also possible. The following approaches are taken to capture the modifications:

—To detect added prefixes or suffixes, one can check if one username is the substring of the other. Thus, we consider the length of the *Longest Common Substring (LCS)* as an informative feature about how similar the username is to prior usernames. We perform a pairwise computation of LCS length between the candidate username and all prior usernames. This will generate a distribution of LCS length values, quantized as features using Equation (7). To get values in range [0,1], we also perform a normalized LCS (normalized by the maximum length of the two strings) and store the distribution as a feature as well.

—For detecting abbreviations, *Longest Common Subsequence* length, is used since it can detect nonconsecutive letters that match in two strings. We perform a pairwise calculation of it between the candidate username and prior usernames and store the distribution as features using aforementioned technique in Equation (7). We also store the normalized version as another distribution feature.

—For swapped letters and added letters, we use the normalized and unnormalized versions of both Edit (Levenshtein) Distance, and Dynamic Time Warping (DTW) [Müller 2007] distance as measures. Again, the end results are distributions, that are saved as features.

Generating Similar Usernames. Users tend to generate similar usernames. The similarity between usernames is sometimes hard to capture using approaches discussed for detecting username modification. For instance, `gateman` and `nametag` are highly similar due to one being the other spelled backward, but their similarity is not recognized by discussed methods. Since we store the alphabet distribution for both the candidate username and prior usernames, we can compare these using different similarity measures. The Kullback–Liebler divergence (KL) [Cover and Thomas 2006] is commonly the measure of choice; however, since KL isn't a metric, comparison among values becomes difficult. To compare distributions, we use the Jensen–Shannon divergence (JS) [Lin 1991], which is computed from KL and is a metric

$$JS(P||Q) = \frac{1}{2}[KL(P||M) + KL(Q||M)], \tag{9}$$

where $M = \frac{1}{2}(P + Q)$, and KL divergence is

$$KL(P||Q) = \sum_{i=1}^{|P|} P_i \cdot log\left(\frac{P_i}{Q_i}\right). \tag{10}$$

Here, P and Q are the alphabet distributions for candidate username and prior usernames. As an alternative, we also consider cosine similarity between the two distributions as a feature. Note that JS divergence does not measure the overlap between the alphabets. To compute alphabet overlaps, we add Jaccard Distance as a feature.

Username Observation Likelihood. Finally, we believe the order in which users juxtapose letters to create usernames depends on their prior knowledge. Given this prior knowledge, we can estimate the probability of observing candidate username. Prior knowledge can be gleaned based on how letters come after one another in prior usernames. In statistical language modeling, the probability of observing username u, denoted in characters as $u = c_1c_2 \dots c_n$, is

$$p(u) = \Pi_{i=1}^{n} p(c_i|c_1c_2 \dots c_{i-1}). \tag{11}$$

We approximate this probability using an n-gram model

$$p(u) \approx \Pi_{i=1}^{n} p(c_i|c_{i-(n-1)} \dots c_{i-1}). \tag{12}$$

Commonly, to denote the beginning and the end of a word special symbols are added: $\star$ and $\bullet$. So, for username sara, the probability approximated using a 2-gram model is

$$p(sara) \approx p(s|\star)p(a|s)p(r|a)p(a|r)p(\bullet|a). \tag{13}$$

To estimate the observation probability of the candidate username using an n-gram model, we first need to compute the probability of observing its comprising n-grams. The probability of observing these n-grams can be computed using prior usernames. These probabilities are often hard to estimate, since some letters never occur after others in prior usernames while appearing in the candidate username. For instance, for candidate username test12 and prior usernames {test, testing}, the probability of $p(\texttt{1}| \star \texttt{test}) = 0$ and therefore $p(\texttt{test12}) = 0$, which seems unreasonable. To estimate probabilities of unobserved n-grams, a smoothing technique can be used. We use the state-of-the-art *Modified Kneser–Ney (MKN)* smoothing technique [Chen and Goodman 1996], which has discount parameters for n-grams observed once, twice, and three times or more. The discounted values are then distributed among unobserved n-grams. The model has demonstrated excellent performance in various domains [Chen and Goodman 1996]. We include the candidate username observation probability, estimated by an MKN-smoothed 6-gram model, as a feature.

We have demonstrated how behavioral patterns can be translated to meaningful features for the task of user identification. These features are constructed to mine information hidden in usernames due to individual behaviors when creating usernames. Overall, we construct 414 features for the candidate username and prior usernames. Figure 5 depicts a summary of these behavioral patterns observed in individuals when selecting usernames.

Clearly, our features do not cover all aspects of username creation, and with more theories and behaviors in place, more features can be constructed. We will empirically study if it is necessary to use all features and the effect of adding more features on learning performance of user identification.

Following MOBIUS methodology, the feature values are computed over labeled data, and the effectiveness of MOBIUS is verified by learning an identification function. Next, experiments for evaluating MOBIUS are detailed.

3.2. Experiments

The MOBIUS methodology is systematically evaluated in this section. First, we verify if MOBIUS can learn an accurate identification function, comparing with some

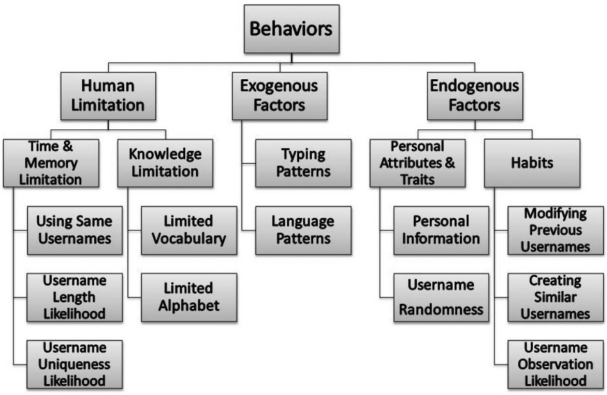


Fig. 5. Individual behavioral patterns when selecting usernames.

baselines. Second, we examine if different learning algorithms make significant difference in learning performance using acquired features. Then, we perform feature importance analysis, and investigate how the number of usernames and the number of features impact learning performance. Before we present our experiments, we detail how experimental data is collected.

3.2.1. Data Preparation. A simple method for gathering identities across social networks is to conduct surveys and ask users to provide their usernames across social networks. This method can be expensive in terms of resource consumption, and the amount of gathered data is often limited. Companies such as Yahoo! or Facebook ask users to provide this kind of information[7]; however, this information is not publicly available.

Another method for identifying usernames across sites is by finding users manually. Users, more often than not provide personal information such as their real names, E-mail addresses, location, gender, profile photos, and age on these websites. This information can be employed to map users on different sites to the same individual. However, manually finding users on sites can be quite challenging.

Fortunately, there exist websites where users have the opportunity of listing their identities (user accounts) on different sites. This can be thought of as *labeled* data for our learning task, providing a mapping between identities. In particular, we find social networking sites, blogging and blog advertisement portals, and forums to be valuable sources for collecting multiple identities of the same user.

Social Networking Sites. On most social networking sites such as Google+ or Facebook, users can list their IDs on other sites. This provides usernames of the same individual on different sites.

Blogging and Blog Advertisement Portals: To advertise their blogs, individuals often join *blog cataloging* sites to list not only blogs, but also their profiles on other sites. For instance, users in BlogCatalog are provided with a feature called “My Communities”. This feature allows users to list their usernames in other social media sites.

Forums: Many forums use generic Content Management Systems (CMS), designed specifically for creating forums. These applications usually allow users to add their usernames on social media sites to their profiles. Examples of these applications that contain this feature include, but are not limited to: vBulletin, phpBB, and Phorum.

[7] http://mashable.com/2010/10/17/y-connect-yahoo/.

Table V. MOBIUS Performance Compared to Content-Based Methods and Baselines

Technique	Accuracy
MOBIUS (Naive Bayes)	91.38%
Method of Zafarani et al. [Zafarani and Liu 2009a]	66.00%
Method of Perito et al. [Perito et al. 2011]	77.59%
Baseline b_1: Exact Username Match	77.00%
Baseline b_2: Substring Matching	63.12%
Baseline b_3: Patterns in Letters	49.25%

We utilize these sources for collecting usernames, guaranteed to belong to the same individual. Overall, 100,179 (c-U) pairs are collected, where c is a username and U is the set of prior usernames. Both c and U belong to the same individual. The dataset contains usernames from 32 sites such as: Flickr, Reddit, StumbleUpon, and YouTube. This dataset contains all the usernames (nodes) collected in Section 2.2.1 as well as additional usernames to make our results comparable.

The collected pairs are considered as positive instances in our dataset. For negative instances, we construct instances by randomly creating pairs (c_i-U_j) such that c_i is from one positive instance and U_j is from a different positive instance ($i \neq j$) to guarantee that they are not from the same individual. We generated different numbers of negative instances (up to 1 million instances), but its effect on the accuracy of learning the identification function was negligible. By further investigation we noticed that this phenomenon takes placed due to feature values for negative instances being far different from that of positive instances. Thus, we continue with a dataset where the class balance is 50% for each label (100,179 positive + 100,179 negative ≈200,000 instances). Then, we compute our 414 feature values for this data and employ this dataset for our learning framework.

3.2.2. Learning the Identification Function. To evaluate MOBIUS, the first step is to verify if it can learn an accurate identification function. Given our labeled dataset where all feature values are calculated, learning the identification function can be realized by performing supervised learning on our dataset. We mentioned earlier that a probabilistic classifier can generalize our binary identification function to a probabilistic one, where the probability of a candidate username belonging to an individual is measured. Probabilistic classification can be achieved by a variety of Bayesian approaches. We select Naive Bayes. Naive Bayes, using 10-fold cross validation, correctly classifies 91.38% of our data instances.

There is a need to compare MOBIUS performance to other content- and link-based methods. To the best of our knowledge, methods from Zafarani and Liu [2009a] and Perito et al. [2011] are the only content-based methods that tackle the same problem with usernames. The ad hoc method of Zafarani et al. employs two features: 1) exact match between usernames and 2) substring match between usernames. Perito et al.'s method uses a single feature. This feature, similar to our username-observation likelihood, utilizes a 5-gram model to compute the username observation probability. Table V reports the performance of these techniques over our datasets. Our method outperforms the method of Zafarani et al. by 38% and the method of Perito et al. by 18%. The key difference between MOBIUS and the methods in comparison is that MOBIUS takes a behavioral modeling approach that systematically generates features for effective user identification.

To evaluate the effectiveness of MOBIUS, we also devise three content-based baseline methods for comparison. When people are asked to match usernames of individuals, commonly used methods are "exact username matching", "substring matching", or finding "patterns in letters". Thus, they form our three baselines b_1, b_2, and b_3:

Table VI. MOBIUS Performance Compared to Link-Based Reference Points

Technique	AUC
MOBIUS (Naive Bayes)	0.937
Reference Point 1: Common Neighbors	0.504
Reference Point 1: Jaccard Coefficient	0.503
Reference Point 1: Adamic/Adar	0.501

Table VII. MOBIUS Performance for Different Classification Techniques

Technique	AUC	Accuracy
J48 Decision Tree Learning	0.894	90.87%
Naive Bayes	0.937	91.38%
Random Forest	0.957	93.59%
ℓ_2-Regularized ℓ_2-Loss SVM	0.950	93.70%
ℓ_1-Regularized ℓ_2-Loss SVM	0.951	93.71%
ℓ_2-Regularized Logistic Regression	0.950	93.77%
ℓ_1-Regularized Logistic Regression	0.951	**93.80%**

b_1: Exact Username Match. It considers an instance positive if the candidate username is an exact match to $\alpha\%$ of the prior usernames. To set α accurately, we computed the percentage of prior usernames that are exact matches to the candidate username in each of our positive instances and averaged it over all positive instances to get α, $\alpha \approx 54\%$. To further analyze the impact, we set $50\% \leq \alpha \leq 100\%$. Among all α values, b_1 does not perform better than 77%.

b_2: Substring Matching. It considers an instance positive if the mean of the candidate username's normalized longest common substring distance to prior usernames is below some threshold θ. We conduct the experiment for the range $0 \leq \theta \leq 1$. In the best case, b_2 achieves 63.12% accuracy.

b_3: Patterns in Letters. For finding letter patterns, b_3 uses the alphabet distribution for the candidate username and the prior usernames as features. Using our data labels, we perform logistic regression. b_3 achieves 49.25% accuracy.

Our proposed technique outperforms baseline b_1, b_2, and b_3 by 19%, 45%, and 86%, respectively. The performance for MOBIUS trained by Naive Bayes, other content-based methods, and baselines are summarized in Table V.

To evaluate MOBIUS against link-based methods, we compare it to well-known unsupervised link prediction methods. As MOBIUS does not use link information, the performance of link-based methods only serve as reference points and no improvement will be reported. The methods included as reference points are *Common Neighbors*, *Jaccard Coefficient*, and *Adamic/Adar* [Liben-Nowell and Kleinberg 2007][8]. Comparison between MOBIUS and the link-based reference points are provided in Table VI. Now, we would like to see if different learning algorithms can further improve the learning performance.

3.2.3. Choice of Learning Algorithm. To evaluate the choice of learning algorithm, we perform the classification task using a range of learning techniques and 10-fold cross validation. The AUCs and accuracy rates are available in Table VII. These techniques have different learning biases, and one expects to observe different performances for the same task. As seen in the table, results are not significantly different among

[8] As our dataset lacks link information, we report the best performances obtained across networks using [Zhang et al. 2014]

Table VIII. MOBIUS Performance for Different Behaviors

Set of Features	Accuracy
I. Human Limitations	87.70
-Limitations in Time and Memory	87.70
—Selecting the Same Username	52.42
—Username Length Likelihood	55.88
—Username Creation Likelihood	60.81
-Knowledge Limitations	51.17
—Limited Vocabulary	51.24
—Limited Alphabet	48.55
II. Exogenous Factors	57.37
-Typing Patterns	57.43
-Language Patterns	51.40
III. Endogenous Factors	93.78
-Personal Information	49.25
-Username Randomness	56.00
-Habits	93.65
—Username Modification	93.64
—Generating Similar Usernames	78.37
—Username Observation Likelihood	48.54

these methods. This shows that when sufficient information is available in features, the user identification task becomes reasonably accurate and is not sensitive to the choice of learning algorithm. In our experiments, ℓ_1-Regularized Logistic Regression is shown to be the most accurate method and thus we use it in the following experiments as the method of choice. The classification employs all 414 features. Designing 414 features and computing their values is computationally expensive. Therefore, we try to empirically determine: 1) whether all features are necessary, and 2) whether it makes *economic* sense to add more features, in Sections 3.2.4 and 3.2.5.

3.2.4. Feature Importance Analysis. Feature Importance Analysis analyzes how important different features are in learning the identification function. First, for each behavior we have identified, we group the respective features and measure their impact on the classification task. That is we only use those features in MOBIUS for classification. We previously provided the hierarchy of these behaviors in Figure 5. For each node in this hierarchy (other than the root), we create a feature set and train MOBIUS using only those features. Table VIII provides the performance of MOBIUS with these feature sets. As shown in the Table, features that describe endogenous factors or human limitations are the most effective for user identification. In terms of human limitations, features that capture limitations in time and memory are most suitable for user identification. Similarly, features that capture typing patterns and habits are most suitable from exogenous and endogenous factors, respectively. Finally, the most effective features for user identification are those that capture users' habits.

This analysis does not show individual features that contribute the most to the classification task. Next, we find these individual features. This can be performed by standard feature selection measures such as Information Gain, χ^2, among others. We utilize *odds-ratios* (logistic regression coefficients) for feature importance analysis and ranking features. The top 10 important features are as follows:

(1) Standard deviation of normalized edit distance between the candidate username and prior usernames,
(2) Standard deviation of normalized longest common substring between the username and prior usernames,
(3) Username observation likelihood,

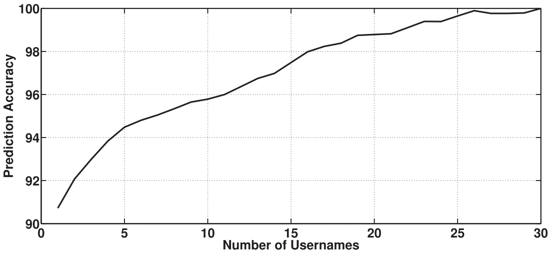


Fig. 6. User identification performance for users with different number of usernames.

(4) Uniqueness of prior usernames,
(5) Exact match: number of times candidate username is seen among prior usernames,
(6) Jaccard similarity between the alphabet distribution of the candidate username and prior usernames,
(7) Standard deviation of the distance traveled when typing prior usernames using the QWERTY keyboard,
(8) Distance traveled when typing the candidate username using the QWERTY keyboard,
(9) Standard deviation of the longest common substring between the username and prior usernames, and
(10) Median of the longest common subsequence between the candidate username and prior usernames.

In fact, a classification using only these 10 features and logistic regression provides an accuracy of 92.72%, which is very close to that of using the entire feature set. We also notice that in our ranked features,

—Numbers `[0-9]` are on average ranked higher than English alphabet letters `[a-z]`, showing that numbers in usernames help better identify individuals, and
—Non-English alphabet letters or special characters, e.g., $\hat{A}, \tilde{A},$ `+`, or `&`, are among the features that could easily help identify individuals across sites, that is, have higher odds-ratios on average.

Although these 10 features perform reasonably well, it is of practical importance to analyze how we can further improve the performance of our methodology in different scenarios, such as by adding usernames or features.

3.2.5. Diminishing Returns for Adding More Usernames and More Features. It is often assumed that when more prior usernames of an individual are known, the task of identifying the individual becomes easier. If true, to improve identification performance, we need to provide MOBIUS with extra prior information (known usernames). In our dataset, users have from one to a maximum of 30 prior usernames. To verify helpfulness of adding prior usernames, we partition the dataset into 30 datasets $\{d_i\}_{i=1}^{30}$, where dataset d_i contains individuals that have i prior usernames. The user identification accuracy on these 30 datasets are shown in Figure 6. We observe a monotonically increasing trend in identification performance, and even for a single prior username, the identification is 90.72% accurate and approaches 100% when 25 or more usernames are available. Note that the identification task is hardest when only a single prior username is available.

Rarely are 25 prior usernames of an individual available across sites. It is more practical to know the minimum number of usernames required for user identification such that further improvements are nominal. The relative performance improvement

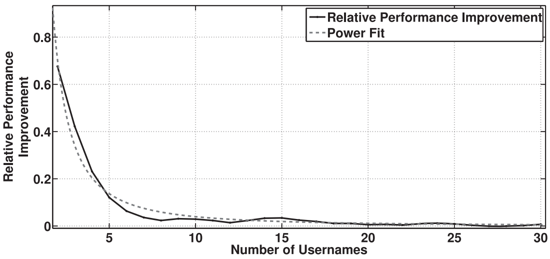


Fig. 7. Relative user identification performance improvement with respect to number of usernames.

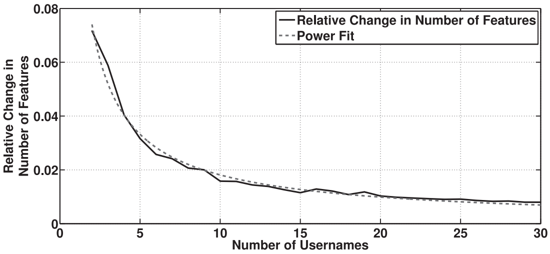


Fig. 8. Relative change in number of features required with respect to number of usernames.

with respect to number of usernames can help us measure this minimum. Figure 7 shows this improvement for adding usernames. We observe a *diminishing return* property, where the improvement becomes marginal as we add usernames and is negligible for more than seven usernames. A power function ($g(x) = 2.44x^{-1.79}$), found with 95% confidence, fits to this curve with adjusted $R^2 = 0.976$. The exponent -1.79 denotes that the relative improvement by adding n usersnames is $\approx 1/n^{1.79}$ times smaller than that by adding a single username, example, for seven usernames, relative identification performance improvement is $\approx$1/33 times smaller than that of a single username.

Similar to adding more prior usernames, one can change number of features. More practically, we would like to analyze how adding features correlates with adding prior usernames. For instance, if we double the number of prior usernames, how many features should we construct (or can be removed) to guarantee reaching a required performance?

To measure this, for each number of prior usernames n, we compute the average number of features such that MOBIUS can achieve fixed accuracy θ. We set θ to the minimum accuracy achievable, independent of number of usernames (90% here). Then we compute the relative change in the number of required features when usernames are added.

Figure 8 plots this relationship. We observe the same diminishing return property, and as one adds more usernames, fewer features are required to achieve a fixed accuracy. A power function ($g(x) = 0.1359x^{-0.875}$), found with 95% confidence, fits to this curve with adjusted $R^2 = 0.987$. The exponent -0.875 denotes that the number of features required for n usersnames is $\approx 1/n^{0.875}$ times smaller than that of a single username.

Finally, if one is left with a set of usernames and a set of features, should we aim at adding more usernames or construct better features? Let $f(n, k)$ denote the performance of our method for n usernames and k features. Let,

$$\delta(n, k) = \frac{f(n+1, k) - f(n, k)}{f(n, k+1) - f(n, k)}. \tag{14}$$

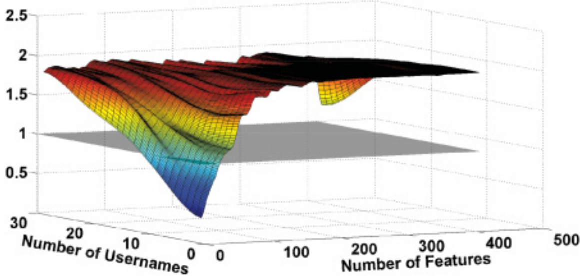


Fig. 9. The $\delta(n, k)$ function, for n usernames and k features. Values larger than 1 show that adding usernames will improve performance more and values smaller than 1 show adding features is better.

The δ function is a finite difference approximation for the derivative ratio with respect to n and k. When $\delta(n, k) > 1$, adding usernames improves performance more and when $\delta(n, k) < 1$, adding features is better. To compute $f(n, k)$, for different values of n, we select random subsets of size k. We denote the average performance over these random subsets as $f(n, k)$. Figure 9 plots the $\delta(n, k)$ function. We plot plane $z = 1$ to better show where adding features is more helpful and where usernames are more beneficial. We observe that for small values of n and k, that is, when fewer usernames and features are available, features help best, but for all other cases adding usernames is more beneficial.

3.3. Discussion

We demonstrated that MOBIUS can exploit information redundancies due to user behaviors to identify individuals across sites. The empirical evaluation shows that MOBIUS is effective in across-site user identification.

Back to our initial questions, although we can tell if a username belongs to a username set, but given a username-set, where can we find the candidate usernames? Furthermore, as MOBIUS operates on usernames, a natural question is if there is additional information available such as location, how we can represent and integrate it into MOBIUS. These are practical questions that need to be answered to complete the task of identification

3.3.1. Finding Candidate Usernames. The candidate username needs to be found using the available tools and information. To most users, unless they have access to the deep or hidden web, the only gateway to find information is the public web and in particular, with tools such as web search engines; therefore, we focus on finding usernames on the public web via web search engines. In our experiments, we had several interesting observations that can lead to finding candidate usernames.

We found that for any two usernames, u_1 and u_2 of the same individual, there is a high chance of co-occurrence of these two in search engine results. To verify this, from our dataset we generated around 100,000 *username-username* pairs $< u_1, u_2 >$ where both u_1 and u_2 belonged to the same individual. We found using Google with query "u_1 u_2" that usernames co-occur in nearly 68% of the cases in web search engine results. This finding suggests that 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 pages are many and long and keyword extraction can be quite tedious and will generate many candidate usernames. Our other observations lead to a solution to mitigate this problem. We will review them first before coming back to a solution to this problem.

Table IX. Profile URLs for Popular Social Media Sites

Site	Profile URL Pattern
YouTube	`http://www.youtube.com/test`
Flickr	`http://www.flickr.com/photos/test`
Reddit	`http://www.reddit.com/user/test`
Del.icio.us	`http://del.icio.us/test`

We observed that for any social media site s and for all its usernames, there exists URLs on the Registered Domain Name of s that contain the username. These URLs are most commonly pointing to the profile/homepage of the users on that site. Denote these URLs as *Profile URLs*. 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 sites in Table IX. We have analyzed 32 online sites in our dataset and surprisingly, in all 32, the site's profile URLs contains the username.

Back to our original problem, interestingly, users often list their other usernames on Profile URLs. For instance, on their profile pages, they list their email addresses, where its part before the @ sign, is a commonly employed username of the individual. In other words, for two usernames u_1 and u_2 of the same individual, it is sufficiently likely for u_1 to exist in the *URL of the webpages* retrieved using popular search engines, such that the page itself contains u_2, that is, u_1 profile page contains u_2.

To verify this, we used our 100,000 *username-username* pairs and for each pair $< u_1, u_2 >$, two separate queries were sent to Google (first username occurring on second username's profile, and vice versa). In Google, the queries can be formulated in the following format: "`inurl:`u_1 u_2" and "`inurl:`u_2 u_1". This phenomenon holds in nearly 38% of the situations. Likewise our previous observation, this suggests that 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.

3.3.2. Adding More Information. MOBIUS can use other types of information that is available on social media sites. In general, we can follow the following procedure to integrate new types of information: (1) determine the behavioral patterns that humans exhibit regarding that information, and (2) construct features to capture information redundancies due to behavioral patterns. For example, we have information beyond username such as individual's *location* that is often available on profile pages. Corresponding to candidate username (c) and prior usernames (U), we have *candidate location* and *prior locations*. One behavioral pattern associated with location is that individuals rarely change their locations. In fact, locations change much less than usernames. Therefore, based on this behavioral pattern, we can have an *exact location match* feature that counts the number of times candidate location is observed among prior locations. One can design additional features to capture similarity between candidate location and prior locations. For example, APIs such as the Google Maps API can be used to convert locations to latitude-longitude pairs and then distances between locations can be used to measure similarity.

As the availability of different types of information varies, such information is not as universally available as usernames. However, we believe more information should help identify users better and further investigation is needed to analyze performance gains due to additional information.

4. RELATED WORK

In this section, we focus on summarizing research related to identifying individuals in social media. We provided a review of directly relevant techniques to our study in Section 3.2. In addition to those, the methods of Iofciu et al. [2011] and Liu et al. [2013] approach the same problem but with extra information. Iofciu et al. utilize tag information in addition to a single username feature and Liu et al. use profile metadata, friendship network information, and content based features. Both methods rely on the availability of information that may not be available on social media. Our method only uses username information across sites.

In addition to these methods, there exists related research about (1) *identifying content produced by an individual on the web* or (2) *identifying individuals in a single social network*.

Identifying Content Authorship. In Amitay et al. [2007], the authors look at the content generation behavior of the same individuals in several collections of documents. Based on the overlap between contributions, they propose a method for detecting pages created by the same individual across different collections of documents. They use a method called detection by compression, where Normalized Compression Distance (NCD) [Cilibrasi and Vitányi 2005] is used to compare the similarity between the documents already known to be authored by the individual and other documents. Author detection has been well discussed in restricted domains. In particular, machine learning and data mining techniques have been employed to detect authors in online messages [Zheng et al. 2006], online message boards [Novak et al. 2004; Abbasi and Chen 2005], blogs [Keogh et al. 2004], and in E-mails [De Vel et al. 2001]. Although, one can think of usernames as the content generated by individuals across sites; however, in content authorship detection, it is common to assume large collections of documents, with thousands of words, available for each user, whereas for usernames, the information available is limited to one word.

User Identification on One Site. Deanonymization[9] is an avenue of research related to identifying individuals on a single site. Social networks are commonly represented using graphs where nodes are the users and edges are the connections. To preserve privacy, an anonymization process replaces these users with meaningless, randomly generated, unique IDs. To identify these masked users, a deanonymization technique is performed. Deanonymization of social networks is tightly coupled with the research in privacy preserving data mining [Agrawal and Srikant 2000] or Identity Theft attacks [Bilge et al. 2009]. In Backstrom et al. [2007], theypresent such process where one can identify individuals in these anonymized networks by either manipulating networks before they are anonymized or by having a priori knowledge about certain anonymized nodes. Narayanan and Shmatikov [2008] present statistical deanonymization technique against high-dimensional data. They argue that given little information about an individual one can easily identify the individual's record in the dataset. They demonstrate the performance of their method by uncovering some users on the Netflix prize dataset using IMDB information as their source for background knowledge. Our work differs from these techniques as it deals with multiple sites. Moreover, it avoids using link information, which is not always available on different social media sites.

5. CONCLUSIONS AND FUTURE WORK

In this article, we have provided empirical evidence on the existence of a mapping between identities of individuals across the social media sites and studied the possibility of identifying users across sites. Both link and content information were used to identify individuals. In the link section, we found that when an individual is present on both networks, there are not relationships between the *number of friends* that the individual has on each network. It was also shown that when the same individual had some friends shared across the two networks, no correlations were observed regarding what percentage of friends on each network were shared. Furthermore, we found that

[9] Deanonymization is tightly coupled with the research in privacy preserving data mining (see [Agrawal and Srikant 2000; Agrawal and Aggarwal 2001; Dinur and Nissim 2003; Evfimievski et al. 2003; Aggarwal and Yu 2008])

the target-node is not very likely to be connected to the crossed-over friends of the base-node, and even in the cases that it is found to be connected, it is challenging to identify it among all connected nodes. These findings and evaluation results of the proposed method show that counter-intuitively, link information is not sufficient to identify individuals across social media sites. However, content information and in particular usernames can be used quite successfully to identify corresponding usernames on various sites. We demonstrated a content-based methodology for connecting individuals across social media sites (MOBIUS). MOBIUS takes a behavioral modeling approach for systematic feature construction and assessment, which allows integration of additional features when required. MOBIUS employs minimal content information available on all social media sites (usernames) to derive a large number of features that can be used by supervised learning to effectively connect users across sites. Users often exhibit certain behavioral patterns when selecting usernames. The proposed behavioral modeling approach exploits information redundancy due to these behavioral patterns. We categorize these behavioral patterns into (1) human limitations, (2) exogenous factors, and (3) endogenous factors. In each category of behaviors, various features are constructed to capture information redundancy. MOBIUS employs supervised learning to connect users. Our empirical results show the advantages of this principled, behavioral modeling approach over earlier methods. The experiments demonstrate that (1) constructed features contain sufficient information for user identification; (2) importance or relevance of features can be assessed, thus features can be selected based on particular application needs; and (3) adding more features can further improve learning performance but with diminishing returns, hence, facing a limited budget, one can make informed decisions on what additional features should be added.

Research issues in this work can be further investigated. As for using link information, we can analyze the likelihood of a target user being connected to friends of the crossed-over friends. As for using content information, future work includes discovering features indigenous to specific sites but beyond those constricted to usernames, and incorporating them into MOBIUS for future needs. Furthermore, hybrid link- and content-based methods are expected to improve the performance of user identification. In addition to improving this user identification accuracy, the behavioral modeling approach of MOBIUS can be used in different domains in social media research. Recently, Rajadesingan et al. showed that this approach is useful for sarcasm detection in social media [Rajadesingan et al. 2015].

Identifying users across social media sites opens the door to many interesting applications. For instance, MOBIUS can help solve problems such as recommending friends and advertising across different networks, analyzing user patterns across them [Zafarani and Liu 2014], and studying user behavior such as migration across social media sites [Kumar et al. 2011].

REFERENCES

A. Abbasi and H. Chen. 2005. Applying authorship analysis to extremist-group web forum messages. *IEEE Intelligent Systems* 20, 5, 67–75.

Charu C. Aggarwal and Philip S. Yu. 2008. A General Survey of Privacy-Preserving Data Mining Models and Algorithms. *Privacy-Preserving Data Mining*. Springer, 11–52.

Dakshi Agrawal and Charu C. Aggarwal. 2001. On the design and quantification of privacy preserving data mining algorithms. In *Proceedings of the 20th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. ACM, Santa Barbara, CA, USA, 247–255.

R. Agrawal, S. Rajagopalan, R. Srikant, and Y. Xu. 2003. Mining newsgroups using networks arising from social behavior. In *Proceedings of the 12th International Conference on World Wide Web*. ACM, Budapest, Hungary, 535.

R. Agrawal and R. Srikant. 2000. Privacy-preserving data mining. *ACM Sigmod Record* 29, 2, 439–450.

E. Amitay, S. Yogev, and E. Yom-Tov. 2007. Serial sharers: Detecting split identities of web authors. In *SIGIR PAN Workshop*. ACM, Amsterdam, Netherlands.

L. Backstrom, C. Dwork, and J. Kleinberg. 2007. Wherefore art thou R3579X?: Anonymized social networks, hidden patterns, and structural steganography. In *WWW*. ACM, Alberta, Canada, 181–190.

L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. 2006. Group Formation in large social networks: Membership, growth, and evolution. In *KDD*. ACM, Philadelphia, USA, 44–54.

L. Bilge, T. Strufe, D. Balzarotti, and E. Kirda. 2009. All your contacts are belong to us: Automated identity theft attacks on social networks. In *WWW*. ACM, Madrid, Spain, 551–560.

S. F. Chen and J. Goodman. 1996. An empirical study of smoothing techniques for language modeling. In *ACL*. Santa Cruz, USA, 310–318.

R. Cilibrasi and P. M. B. Vitányi. 2005. Clustering by compression. *IEEE Transactions on Information Theory* 51, 4, 1523–1545.

Private Communication. 2013. Private Communication with a Yahoo! employee.

Gordon V. Cormack, José María Gómez Hidalgo, and Enrique Puertas Sánz. 2007. Feature engineering for mobile (SMS) spam filtering. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Amsterdam, Netherlands, 871–872.

T. M. Cover and J. A. Thomas. 2006. *Elements of Information Theory*. Wiley-interscience.

D. Cowan. 1958. *An Introduction to Modern Literary Arabic*. Vol. 240. Cambridge University Press.

D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri. 2008. Feedback effects between similarity and social influence in online communities. In *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, Las Vegas, Nevada, USA, 160–168.

O. De Vel, A. Anderson, M. Corney, and G. Mohay. 2001. Mining e-mail content for author identification forensics. *ACM Sigmod Record* 30, 4, 55–64.

I. Dinur and K. Nissim. 2003. Revealing information while preserving privacy. In *Proceedings of the 22nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. San Diego, California, 202–210.

C. Doctorow. 2012. Preliminary Analysis of LinkedIn User Passwords. http://bit.ly/L5AHo3 (Last accessed October 1, 2015).

T. Dunning. 1994. *Statistical Identification of Language*. CR Lab, New Mexico State University.

D. Easley and J. Kleinberg. 2010. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press.

A. Evfimievski, J. Gehrke, and R. Srikant. 2003. Limiting privacy breaches in privacy preserving data mining. In *Proceedings of the 22nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. San Diego, California, 211–222.

C. A. Ferguson. 1957. Word stress in persian. *Language* 33, 2, 123–135.

Z. Gyongyi, H. Garcia-Molina, and J. Pedersen. 2004. Combating web spam with trustrank. In *Proceedings of the 13th International Conference on Very Large Data Bases-Vol. 30*. VLDB Endowment, 576–587.

N. Habash, A. Soudi, and T. Buckwalter. 2007. On arabic transliteration. *Arabic Computational Morphology*. Springer, 15–22.

T. Iofciu, P. Fankhauser, F. Abel, and K. Bischoff. 2011. Identifying users across social tagging systems. In *ICWSM*. AAAI, Barcelona, Spain, 522–525.

E. Keogh, S. Lonardi, and C. A. Ratanamahatana. 2004. Towards parameter-free data mining. In *KDD*. ACM, Seattle, WA, 206–215.

G. Kossinets and D. J. Watts. 2006. Empirical analysis of an evolving social network. *Science* 311, 5757, 88.

Shamanth Kumar, Reza Zafarani, and Huan Liu. 2011. Understanding user migration patterns in social media. In *AAAI*. San Francisco, CA, 1204–1209.

J. Leskovec, L. Backstrom, R. Kumar, and A. Tomkins. 2008. Microscopic evolution of social networks. In *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, Las Vegas, Nevada, USA, 462–470.

David Liben-Nowell and Jon Kleinberg. 2007. The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology* 58, 7 (2007), 1019–1031.

J. Lin. 1991. Divergence measures based on the shannon entropy. *IEEE Transaction on Information Theory* 37, 1, 145–151.

J. Liu, F. Zhang, X. Song, Y. Song, C. Lin, and H. Hon. 2013. What's in a name?: An unsupervised approach to link users across communities. In *WSDM*. ACM, Rome, Italy, 495–504.

Filippo Menczer. 2007. Web crawling. *Web Data Mining, Exploring Hyperlinks, Contents and Usage Data*. Springer, 273–321.

G. A. Miller. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychol. Rev.* 63, 2, 81–97.

A. Mislove, A. Post, P. Druschel, and K. P. Gummadi. 2008. Ostra: Leveraging trust to thwart unwanted communication. In *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*. USENIX Association, San Fransisco, California, 15–30.

M. Müller. 2007. *Information Retrieval for Music and Motion*. Vol. 6. Springer Berlin.

A. Narayanan and V. Shmatikov. 2008. Robust de-anonymization of large sparse datasets. In *IEEE SSP*. Oakland, CA, 111–125.

J. Novak, P. Raghavan, and A. Tomkins. 2004. Anti-aliasing on the web. In *WWW*. ACM, New York, NY, USA, 30–39.

A. Paivio. 1983. The empirical case for dual coding. *Imagery, Memory and Cognition*. Lawrence Erlbaum Assoc Incorporated, 307–332.

Daniele Perito, Claude Castelluccia, Mohamed Kaafar, and Pere Manils. 2011. How unique and traceable are usernames?. In *PETS*. Waterloo, Canada, 1–17.

Ashwin Rajadesingan, Reza Zafarani, and Huan Liu. 2015. Sarcasm detection on Twitter: A behavioral modeling approach. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining (WSDM'15)*. ACM, New York, NY, USA, 97–106. DOI:http://dx.doi.org/10.1145/2684822.2685316

Sam Scott and Stan Matwin. 1999. Feature engineering for text classification. In *ICML*, Vol. 99. Bled, Slovenia. J. Stefan Institute (IJS), 379–388.

Jie Tang, Tiancheng Lou, and Jon Kleinberg. 2012b. Inferring social ties across heterogenous networks. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining*. ACM, Seattle, Washington, 743–752.

Lei Tang, Huan Liu, and Jianping Zhang. 2012a. Identifying evolving groups in dynamic multimode networks. *IEEE Transactions on Knowledge and Data Engineering* 24, 1, 72–85. DOI:http://dx.doi.org/10.1109/TKDE.2011.159

Lei Tang, Xufei Wang, and Huan Liu. 2012c. Community detection via heterogeneous interaction analysis. *Data Mining and Knowledge Discovery*, Springer, 1–33.

N. Tran, B. Min, J. Li, and L. Subramanian. 2009. Sybil-resilient online content voting. In *Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation*. USENIX Association, Boston, Massachusetts, 15–28.

D. J. Watts and S. H. Strogatz. 1998. Collective dynamics of small-world networks. *Nature* 393, 6684, 440–442.

Wikipedia. 2015. Keyboard Layouts. http://bit.ly/kXso (Last accessed October 1, 2015).

J. Yan, A. Blackwell, R. Anderson, and A. Grant. 2000. The memorability and security of passwords-some empirical results. *U. of Cambridge Tech. Rep*.

H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman. 2006. Sybilguard: Defending against sybil attacks via social networks. In *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*. ACM, Pisa, Italy, 267–278.

Reza Zafarani, Mohammad Ali Abbasi, and Huan Liu. 2014. *Social Media Mining: An Introduction*. Cambridge University Press.

Reza Zafarani and Huan Liu. 2009a. Connecting corresponding identities across communities. In *ICWSM*. AAAI, San Jose, California, 354–357.

Reza Zafarani and Huan Liu. 2009b. Social computing data repository at ASU. *School of Computing, Informatics and Decision Systems Engineering, Arizona State University*.

Reza Zafarani and Huan Liu. 2013. Connecting users across social media sites: A behavioral-modeling approach. In *KDD*. ACM, Chicago, Illinois.

Reza Zafarani and Huan Liu. 2014. Users joining multiple sites: Distributions and patterns. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*. AAAI, Ann Arbor, MI, USA.

Jiawei Zhang, Xiangnan Kong, and Philip S. Yu. 2014. Transferring heterogeneous links across location-based social networks. In *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*. ACM, New York, NY, USA, 303–312.

R. Zheng, J. Li, H. Chen, and Z. Huang. 2006. A framework for authorship identification of online messages: Writing-style features and classification techniques. *JASIST* 57, 3, 378–393.

X. Zhu. 2005. *Semi-Supervised Learning Literature Survey.*Technical Report 1530. *Computer Sciences, University of Wisconsin-Madison*.

Received February 2014; revised January 2015; accepted March 2015