Identifying Your Customers in Social Networks

Chun-Ta Lu
University of Illinois at Chicago
clu29@uic.edu

Hong-Han Shuai
National Taiwan University
d99942020@ntu.edu.tw

Philip S. Yu
University of Illinois at Chicago
psyu@cs.uic.edu

ABSTRACT

Personal social networks are considered as one of the most influential sources in shaping a customer's attitudes and behaviors. However, the interactions with friends or colleagues in social networks of individual customers are barely observable in most e-commerce companies. In this paper, we study the problem of customer identification in social networks, i.e., connecting customer accounts at e-commerce sites to the corresponding user accounts in online social networks such as Twitter. Identifying customers in social networks is a crucial prerequisite for many potential marketing applications. These applications, for example, include personalized product recommendation based on social correlations, discovering community of customers, and maximizing product adoption and profits over social networks.

We introduce a methodology CSI (Customer-Social Identification) for identifying customers in online social networks effectively by using the basic information of customers, such as username and purchase history. It consists of two key phases. The first phase constructs the features across networks that can be used to compare the similarity between pairs of accounts across networks with different schema (e.g. an e-commerce company and an online social network). The second phase identifies the top-K maximum similar and stable matched pairs of accounts across partially aligned networks. Extensive experiments on real-world datasets show that our CSI model consistently outperforms other commonly-used baselines on customer identification.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications – Data Mining

Keywords

E-commerce; Social Network; Customer Identification

1. INTRODUCTION

Personal social networks affect the adoption of individual innovations and products [13]. For example, customers usually gather information from friends, when they contemplate purchasing goods and services. Customers also share opinions within their social networks regarding to different products which they have recently purchased or they are familiar with. Such actions of acquiring and disseminating information are critical to understanding customer behaviors and analyzing the factors that affect a customer's decisions [16]. However, these actions are implicit in the social connections (e.g., the relationship of friends or colleagues) that are barely observable in most e-commerce sites.

Fortunately, the emergence of online social networks, such as Twitter and Facebook, presents a great opportunity to access publicly available information of social connections. It appears that considerable potential exists for novel applications via leveraging the rich information from online social networks. Examples of applications include prediction of product adoption [6], personalized product recommendation via exploiting social correlation [6, 9] , and maximization of product adoption and profits over social networks [5]. In addition, mining the integrated information from social networks and commercial companies leads to other promising applications, such as discovering community of customers and analyzing opinions [15, 28] of target customer communities for designing a marketing strategy. One common and crucial assumption of these applications is the knowledge of social connections between customers. However, since an online social network is built for social communication, this knowledge has not been used for e-commerce.

To fulfill the gap between conventional companies and social networks, in this paper, we tackle the problem of customer identification in social networks. The mapping of customers to their user accounts in social networks serves as a prerequisite for applying existing marketing techniques to a broader range of e-commerce. Moreover, astroturfing becomes a serious problem in e-commerce nowadays. In 2013, the survey conducted by Dimensional Research[1] shows that 90% of consumers are influenced by online reviews in their purchasing decisions. The false advertising not only influences a large amount of customers to make wrong purchasing decisions but also slanders good products/companies. However, it is challenging to verify whether the review is spam or not due to lack of user information. Therefore, identifying customers in online social networks also provides a promising way to facilitate fake review detection[2].

CIKM'14, November 3–7, 2014, Shanghai, China.
Copyright 2014 ACM 978-1-4503-2598-1/14/11 ...$15.00.
http://dx.doi.org/10.1145/2661829.2662057.

[1] http://www.zendesk.com/resources/customer-service-and-lifetime-customer-value

[2] The privacy issues are worth discussing. According to the Consumer Privacy Bill of Rights, e-commerce sites should

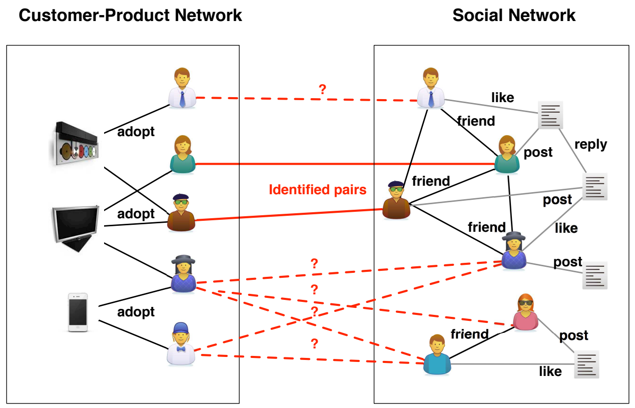


Figure 1: Example of customer identification across a customer-product network and a social network.

Generally, in an e-commerce system, customers interact with products (or services) only[3], while users in an online social network have connections with each other and interact with user-generated contents (e.g., tweets, pictures and videos posted by users). Therefore, the schema of these two systems are essentially different: the former is a bipartite customer-product network, but the latter is a general heterogeneous social network involving all kinds of connections among users and user-generated contents.

Figure 1 shows an example of a customer-product network and a social network. In the customer-product network, five customers adopt three products; meanwhile, six users discuss these products in the social network. Note that among the five customers, four of them also have user accounts in the social network but only two customers are identified (pairs of accounts marked in solid red lines). The task of customer identification is to discover which pair of accounts, as shown in question marks in Figure 1, belongs to the same idenvidual in real-world.

Although users may create alias accounts on social networks, in most cases users will stick to a single account because of the difficulty of managing multiple accounts. Furthermore, only the primary account that reveals the major social activities is of interest to the investigation. Hence, we assume that each customer shall be identified as at most one (primary) user account in social network and vise versa[4].

Despite its value and significance, the customer identification task has not been addressed as it is very challenging due to the following two reasons:

1) *Difference in network schema.* Unlike most prior works on link prediction [12, 20, 21, 18], customer identification requires to predict links across networks with completely different schema (i.e., bipartite network vs. general heterogeneous network). Most existing features for link prediction, such as number of common neighbors and Jaccard's coefficient, are computed by enumerating the connections between nodes within a single network. However, due to the one-to-one constraint on the links across multiple networks, existing features will reduce to a constant value if we directly apply them to predict links across networks [18]. The situation is even more severe when one of the networks is a bipartite network, where no connections exist between customers. Although a bipartite network can be projected onto a unimodal network [4], such as a co-adoption network, many important features (e.g., interests of customers) will be lost during the transformation. Furthermore, customers barely have social interactions with neighbors in the unimodal network [10].

2) *Partially aligned networks.* Another fundamental problem lies in the fact that most networks can only be partially aligned, w.r.t the one-to-one constraint. For example, in Figure 1, not all customers have accounts in the social network. Thus, anchor link prediction [18] and conventional network alignment approaches [3], which assume that two networks are fully aligned, cannot be directly used in the customer identification problem. A detailed comparison between customer identification problem and other related problems are reported in Table 1.

To tackle the customer identification problem involving the above issues, we present the following contributions:

- We formulate the customer identification problem and present the problem analysis. To the best of our knowledge, our work is the first to focus on connecting users between e-commerce companies and online social networks (Section 2).
- Our approach, called CSI (Customer-Social Identification), can be applied to most e-commerce companies by using the basic information of customers, such as username and purchase history. To compare the similarity between users across networks, we transform existing social features for link prediction into heterogeneous features, e.g., common interests of users across networks (Section 3.1).
- We propose to formulate the multi-network partial alignment problem as a top-K maximum similarity and stable matching problem. Based on scores of similarity, CSI method can effectively identify customers in social network w.r.t one-to-one constraint (Section 3.2).
- Through extensive experiments on real-world datasets spanning 10 months, we demonstrate that CSI method consistently outperforms other commonly-used baselines – with up to 38% improvement on F1-score and 21% improvement on AUC (Section 4).

We discuss related work in Section 5 and conclude with possible extensions in Section 6.

2. PROBLEM FORMULATION

The customer identification problem we focus on, in this paper, is to connect customer accounts at an e-commerce site (represented as a customer-product network) to the corresponding user accounts in an online social network. Though the proposed framework can easily be generalized to the setting with more than one pair of networks. In this section, we first define the concept of customer-product network and social network, and then present the formulation of the customer identification problem. Table 2 lists the main notations we use throughout the paper.

provide the privacy settings that allow users to avoid being tracking and keep their feedbacks/reviews private. On the other hand, users are encouraged and have better to adjust the privacy settings to their comfort levels.

[3] Although contents generated by customers are useful, they are rare in most commercial companies, and thus they are not included in this work.

[4] We ignore the case that an individual can have multiple accounts in the same network which is a different research problem and has been addressed in [7].

Table 1: Summary of related problems

Property	**Customer Identification**	**Anchor Link Prediction**[18]	**Network Alignment** [3]	**User Profile Matching** [31, 27]	**Relational Entity Resolution** [7]	**Link Prediction** [12, 20, 21]
target link relationship	one-to-one	one-to-one	one-to-one	one-to-one	clustering	many-to-many
target link type	inter-network	inter-network	inter-network	inter-network	intra-network	intra-network
#network	multiple	multiple	multiple	multiple	single	single/multiple
network schema	bipartite vs. heterogeneous[a]	heterogeneous	homogenous	heterogeneous	homogenous/ heterogeneous	homogenous/ heterogeneous
target network relationship	partially aligned	fully aligned	fully aligned	N/A	N/A	N/A

[a] Bipartite network is a specific heterogenous network, whose nodes are divided into two disjoint sets.

Customer-Product Network: Let $\mathcal{G}^c = (\mathcal{U}^c, \mathcal{P}, \mathcal{E}^c)$ denote a customer-product network, where $\mathcal{U}^c$ is the set of customers, $\mathcal{P}$ is the set of products, and $\mathcal{E}^c \subset \mathcal{U}^c \times \mathcal{P}$ is the set of *adoption links*. The type of adoption, depending on the genre of the e-commerce site, can be purchase of a product, subscription of a video or check-in on a hotel. To provide a general model for most e-commerce sites, we consider only the structure properties between customers and products and discard the semantic meaning of the adoption.

In online social networks, a large amount of contents is generated by users daily and most of them are irrelevant to the concerns. For the sake of efficiency, one may filter out redundant messages by setting predefined rules. For instance, an e-commence company can specify a list of terms related to the products of interest to the company and inquire for the relevant contexts from online social networks. Therefore, the user-generated contents in the social network after filtering should be relevant to the products of interests, e.g., either containing the names of the products or the URL links to the product pages in the e-commerce site. Without loss of generality, we assume the customer-product network and the social network share the same sets of products of interests P. Here we focus on studying the social networks filtered with the product related terms.

Social Network: A social network is represented as $\mathcal{G}^s = (\mathcal{V}^s, \mathcal{E}^s)$, where $\mathcal{V}^s = \mathcal{U}^s \bigcup \mathcal{P}$ is the set of nodes including two types of nodes. $\mathcal{U}^s$ is the set of users and $\mathcal{P}$ is the set of the products of interests mentioned in the user-generated contents. $\mathcal{E}^s \subset \mathcal{V}^s \times \mathcal{V}^s$ is the set of edges of different types in the social network $\mathcal{G}^s$. The types of edges include the social links between users, the links between users and the products mentioned by the users, represented by $\mathcal{U}^s \times \mathcal{U}^s$ and $\mathcal{U}^s \times \mathcal{P}$, respectively.

Customer Identification: Suppose we have a customer-adoption network $\mathcal{G}^c$ and a social network $\mathcal{G}^s$, with a small set of identified pairs $\mathcal{A}$, the task of customer identification is to find the optimal set $\mathcal{A}^*$ in which all the customers in $\mathcal{G}^c$, who can be identified in $\mathcal{G}^s$, are matched to their corresponding accounts in $\mathcal{G}^s$.

Given a candidate pair (u_i^c, u_j^s) of a customer u_i^c in $\mathcal{U}^c$ and a social network user u_j^s in $\mathcal{U}^s$, we shall decide whether this pair belongs to the same individual. Let $f(u_i^c, u_j^s)$ denote the *customer identification function*, i.e.,

$$f(u_i^c, u_j^s) = \begin{cases} 1, & \text{if } u_i^c \in \mathcal{U}^c,\ u_j^s \in \mathcal{U}^s \text{ and} \\ & (u_i^c, u_j^s) \text{ belong to the same individual,} \\ 0, & \text{otherwise.} \end{cases}$$

Table 2: Notation Summary

Symbol	Definition and Description
$\mathcal{G}^s$	network $\mathcal{G}^c = (\mathcal{U}^c, \mathcal{P}, \mathcal{E}^c)$, $\mathcal{G}^s = (\mathcal{V}^s, \mathcal{E}^s)$
$\mathcal{V}^s$	set of nodes in $\mathcal{G}^s$, $\mathcal{V}^s = \mathcal{U}^s \bigcup \mathcal{P}$
$\mathcal{U}^s$	set of users in $\mathcal{G}^s$
$\mathcal{P}$	set of products in both $\mathcal{G}^c$ and $\mathcal{G}^s$
$\mathcal{E}^s$	set of edges in $\mathcal{G}^s$
$\mathcal{A}$	set of the identified pairs across networks. $\mathcal{A}^*$ is the optimum set
$\Gamma(v_i^s)$	neighbors of the node v_i in $\mathcal{G}^s$
$\Gamma_u(u_i^s)$	friends of the user u_i in $\mathcal{G}^s$
$\Gamma_p(u_i^s)$	products that link to user u_i in $\mathcal{G}^s$
$\Gamma_u^s(p_x)$	users that link to the product p_x in $\mathcal{G}^s$
u_i^s	user in $\mathcal{G}^s$
p_x	product in $\mathcal{G}^c$ and $\mathcal{G}^s$
(u_i^c, u_j^s)	candidate pair across networks
$f(u_i^c, u_j^s)$	customer identification function
$score(u_i^c, u_j^s)$	similarity score of candidate pair (u_i^c, u_j^s)

Recall that each customer can only be identified as at most one (primary) user account in a social network and vise versa, i.e., one-to-one constraint. Hence, the set of known pairs $\mathcal{A}$ can be defined in the following formula:

$$\begin{aligned} \mathcal{A} = \{(u_i^c, u_j^s) | f(u_i^c, u_j^s) = 1, \text{and} \\ \nexists u_{i'}^c, u_{j'}^s, \text{s.t. } f(u_{i'}^c, u_j^s) = 1 \text{ or } f(u_i^c, u_{j'}^s) = 1\} \end{aligned}$$

, where $i \neq i'$ and $j \neq j'$. The optimum set $\mathcal{A}^*$ is the maximum set of $\mathcal{A}$, since $\mathcal{A}^*$ contains all the customers who can be identified in the social network. In addition, due to the one-to-one constraint, $\mathcal{A}^*$ is unique, i.e., no other combination of pairs that have the same size as $\mathcal{A}^*$.

The customer identification task serves as a prerequisite for developing many potential marketing applications in general e-commerce sites, as we have discussed in the Introduction. However, it involves two key challenges that make it difficult to be solved by applying existing social link prediction techniques [12, 20, 21, 18]. First, the target links to be predicted are one-to-one relationships between two sets of nodes across networks with completely different schema (e.g., a customer-product network and a social network). To predict the existence of target links, we shall compare the similarity between pairs of nodes across networks. However, most existing features for link prediction, such as number of common neighbors, are designed for predicting the target links within a single network. The social features that exploit the social connections of identified pairs across networks are also not applicable, since there are no connections between customers in customer-product networks.

How can we extract informative features for this customer identification task using basic information available in most e-commerce sites? Second, the prediction of all target links should be considered collectively, not only due to the one-to-one constraint but, more importantly, because the nature of multiple networks tends to be partially aligned. How can we effectively match all the customers, who can be identified in social networks, to their corresponding social user accounts?

3. CUSTOMER-SOCIAL IDENTIFICATION

In this section, we introduce a novel method, CSI (Customer-Social Identification), for effectively identifying customers in social networks. It consists of two phases, each of which addresses one major challenge of customer identification. The first phase tackles the feature extraction across networks with different schema, while the second phase manages to identify customers in partially aligned networks.

3.1 Extracting features across networks with different schema

As the first phase, CSI constructs the features that can be used to measure the similarity between pairs of accounts across networks with different schema. Because individuals often exhibit consistent behavioral patterns across networks, such as selecting similar usernames and passwords [29, 31, 22], we can make use of the similarities between candidate pairs to discover the same individuals.

Considering our purpose is to provide a general model for most e-commerce sites, we shall extract features by using the basic customer information which is generally available. Therefore, two common information sources are investigated: user profiles and the (product) interests of users. In the following, we present several similarity measures corresponding to each information source. The scores of these measures will be treated as the features for the next phase.

3.1.1 Modeling user profile similarity

When a customer registers an account in an e-commerce site, s/he is usually asked to select a unique username and to fill in her/his full name and email address. This registration builds up the basic user profile of the customer. Other attributes, such as the city of residency, gender and ages, are also useful to identify individuals. Though, these attributes are inconsistent in multiple sites and often left blank by the customer. Hence, we attempt to measure the similarity mainly by exploiting names and email addresses.

Names: Usernames are unique on each web site and can be viewed as identifiers of individuals, whereas the full names, i.e., the combinations of first name and last name, are not unique. In [31], Zafarani et. al. observed that human tends to have consistent behavior patterns when selecting usernames in different social media sites. For example, individuals often select new usernames by changing their previous usernames, such as add prefixes or suffixes or abbreviate part of their full names. However, their study mainly focus on the assumption that multiple prior usernames of the same individuals are available. This may not be an appropriate assumption in our problem, because most e-commerce sites usually obtain only one single prior username of each customer.

Therefore, among the top 10 important features presented in [31], we select the four features that can be calculated by the single prior username. Besides, we also consider the Levenshtein Edit Distance [19], which can capture the changes of candidate usernames, as another feature. The five features are listed as follows:

- Exact username match,
- Jaccard similarity between the alphabet distribution of the candidate username and the prior username,
- Distance traveled when typing the candidate username using the QWERTY keyboard,
- Longest common subsequence between the candidate username and the prior username,
- Levenshtein edit distance.

Email: Email addresses can uniquely identify individuals, whereas they are not public available in most online social networks. In this paper, email addresses are used as for verification of the identification. Once we discover that they exist in both customer profiles and user profiles in online social networks, we can pair the both accounts of their owners and put them into the set of identified pairs.

3.1.2 Modeling user interest similarity

In additional, the products that adopted by customers and mentioned by social network users reflect their common interests to some extent. Therefore, we propose to extract user interest features based on the similarity between the products of interests that customers and social network users both have in common.

Here we extend the definition of some of the most effective measures in social link prediction [20, 1, 32]. All the measures compute the similarity between customer u_i^c in $\mathcal{G}^c$ and social network user u_j^s in $\mathcal{G}^s$, and assign a similarity $score(u_i^c, u_j^s)$ to the candidate pair (u_i^c, u_j^s).

1) **Common Interests (CI):** The most direct implementation of this idea for customer identification is to consider the number of interests that customer u_i^c and social network user u_j^s both have in common. We denote the interests of u_i^c as $\Gamma_p(u_i^c)$ and the interests of u_j^s as $\Gamma_p(u_j^s)$. The score of common interests is defined as follows:

$$\begin{aligned} score(u_i^c, u_j^s) &= |\{p_x|(u_i^c, p_x) \in \mathcal{E}^c\} \cap \{p_y|(u_j^s, p_y) \in \mathcal{E}^s\}| \\ &= |\Gamma_p(u_i^c) \cap \Gamma_p(u_j^s)| \end{aligned} \tag{1}$$

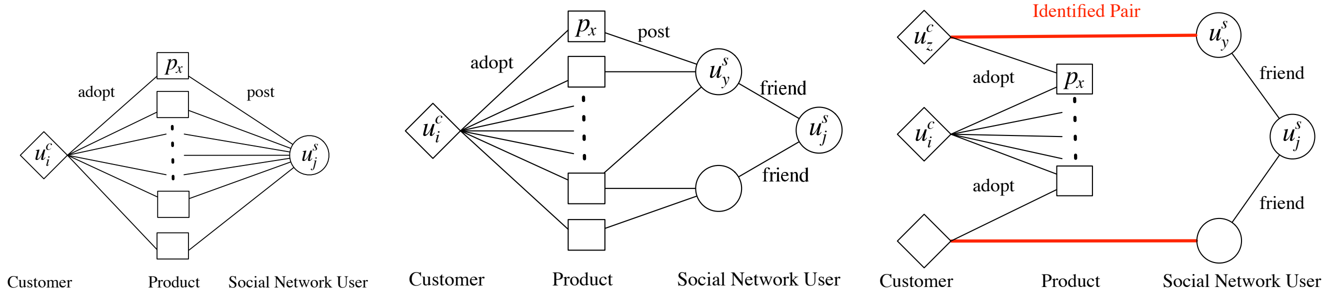
where $|\mathcal{P}|$ is the cardinality of the set $\mathcal{P}$.

2) **Jaccard's Coefficient (JC):** The Jaccard's coefficient is a normalized version of common interests, i.e., the number of common interests divided by the total number of distinct products of interests in $\Gamma_p(u_i^c) \cup \Gamma_p(u_j^s)$.

$$score(u_i^c, u_j^s) = \frac{|\Gamma_p(u_i^c) \cap \Gamma_p(u_j^s)|}{|\Gamma_p(u_i^c) \cup \Gamma_p(u_j^s)|} \tag{2}$$

3) **Admic/Adar Index (AA) [1]:** The AA index refines the simple counting of common interests by weighting rarer interests more heavily. We denote the customers who adopt p_x as $\Gamma_u^c(p_x)$ and the social network users who mention p_x as $\Gamma_u^s(p_x)$. We extend the AA index into multi-network settings, where the common interests are weighted by their average degrees in log scale. The similarity score of u_i^c and u_j^s is derived as follows:

$$\begin{aligned} &score(u_i^c, u_j^s) = \\ &\sum_{\forall p_x \in \Gamma_p(u_i^c) \cap \Gamma_p(u_j^s)} log^{-1}(\frac{|\Gamma_u^c(p_x)| + |\Gamma_u^s(p_x)}{2})| \end{aligned} \tag{3}$$



(a) common interests (b) common interests with friends (c) common interests with identified friends

Figure 2: Modeling user interest similarity

4) **Resource Allocation Index (RA) [32]:** The RA index is similar to the AA index except the weight is distributed averagely instead of in log scale.

$$score(u_i^c, u_j^s) = \sum_{\forall p_x \in \Gamma_p(u_i^c) \cap \Gamma_p(u_j^s)} \left(\frac{|\Gamma_u^c(p_x)| + |\Gamma_u^s(p_x)|}{2}\right)^{-1} \quad (4)$$

Above four measures compute the similarity between customer u_i^c and social network user u_j^s based on their shared (products of) interests directly, as illustrated in Figure 2(a). However, customer u_i^c may not actively mention the products that s/he has adopted in social networks. To compute the interest similarity between u_i^c and u_j^s, we need to seek other connections or paths between them.

According to the researches of social influence on purchase behaviors [14, 13, 6], a customer is more likely to buy a product if his/her friends have widely adopt it. Thus, we consider utilizing the interests of friends to help locate the inactive customers. There are two types of paths between u_i^c and u_j^s through the interests of friends we can exploit. Figure 2(b) shows an example of the first type of a path. In Figure 2(b), the product p_x mentioned by u_y^s, a friend of u_j^s, is also adopted by u_i^c. If u_i^c and u_j^s belong to the same individual, this path $\langle u_i^c, p_x, u_y^s, u_j^s \rangle$ would imply the adoption of p_x is related to the post from u_y^s. The second type of a path is similar to the first one, except this time we will make use of the identified pairs. For example, in Figure 2(c), the product p_x adopted by u_z^c, who is identified as u_y^s (a friend of u_j^s), is also adopted by u_i^c. Similar to the first case, this path $\langle u_i^c, p_x, u_z^c(u_y^s), u_j^s \rangle$ also imply the adoption of p_x made by u_j^s is related to that made by $u_y^s(u_z^c)$, if u_i^c and u_j^s belong to the same individual.

Note that the common interests with (identified) friends is a weaker indicator than the common interests for a candidate pair. In this paper, we extend the *Katz's* index [17] to provide a weighted measure on the collection of paths between u_i^c and u_y^s.

5) **Katz's Index [17]:** The Katz's index sums over the collection of paths, exponentially damped by length to count short paths more heavily, leading to the β-parameterized measure.

$$score(u_i^c, u_j^s) = \sum_{l=1}^{l_{max}} \beta^l \cdot |paths_{u_i^c, u_j^s}^{\langle l \rangle}| \quad (5)$$

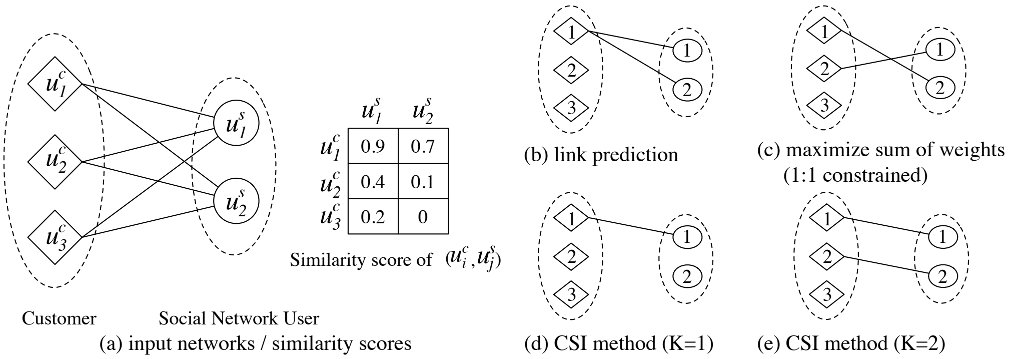
where $paths_{u_i^c, u_j^s}^{\langle l \rangle}$ is the set of all length-l paths from u_i^c to u_j^s. Here we adapt the truncated Katz score, in which the length-l is limited to l_{max} instead of ∞ as in the original Katz's measure, since the truncated Katz often outperforms Katz for link prediction [24]. In this paper, we set $l_{max} = 2$ to capture both factors of the common interests and common interests with (identified) friends. $|paths_{u_i^c, u_j^s}^{\langle 1 \rangle}|$ is the same as the number of common interests, while $|paths_{u_i^c, u_j^s}^{\langle 2 \rangle}|$ is the number of paths through the interests of friends. For example, there are 5 paths between u_i^c and u_j^s in Figure 2(b) and 2 paths between them in Figure 2(c), and thus $|paths_{u_i^c, u_j^s}^{\langle 2 \rangle}| = 5 + 2 = 7$.

3.2 Identifying customers in partially aligned networks

With the features extracted in the previous phase, we can train a binary classifier (e.g., SVM or logistic regression) to roughly decide whether candidate pairs across networks belong to the same identities or not. However, the predictions of the binary classifier cannot be directly used for customer identification. This is because the inference of conventional classifiers are designed for constraint-free settings (e.g., one customers can be paired with multiple user accounts in a social network), and thus the one-to-one constraint on account pairs across networks may not hold.

Instead of simply relying on the decision made by the classifier, we notice that most classifiers also generate similarity scores for classification. Based on the similarity scores that are further calibrated [30], one may think of applying conventional matching techniques, such as stable marriage [11] and maximum weight matching, to find a one-to-one matching between pairs of accounts across two networks. Nevertheless, these techniques could be problematic in the customer identification task, since they usually assume networks are fully aligned, whereas in fact most networks are partially aligned. That is to say, some customers in an e-commerce site do not have any user accounts in an online social network. We should not pair these customers to any user accounts in the social network recklessly; otherwise, we may waste valuable resources on inappropriate targets.

In order to tackle the above issues, we propose to formulate the customer identification in partially aligned networks as a top-K maximum similarity and stable matching prob-



	u^s_1	u^s_2
u^c_1	0.9	0.7
u^c_2	0.4	0.1
u^c_3	0.2	0

Similarity score of (u^c_i, u^s_j)

(a) input networks / similarity scores

(b) link prediction

(c) maximize sum of weights (1:1 constrained)

(d) CSI method (K=1)

(e) CSI method (K=2)

Figure 3: Example of customer identification by different methods. (a) is the input networks and similarity scores, (b) and (c) are the results of different methods. (d) and (e) are the results of CSI methods for $K = 1$ and $K = 2$, respectively.

lem[5]. Specifically, our goal is to find the top K pairs that have the maximum similarity (or weight) among all the stable matching of any combination of K pairs across networks.

Generally speaking, stable matching is a one-to-one matching A with the principle that there is no unmatched pair (m, w) such that m and w both prefer each other to their current assignments in A. Here we say "m prefers w over z", if the score of pair (m, w) is larger than the score of pair (m, z). The primal reason for limiting our solution to stable matching is because stable matching methods can maximize the local benefits of one set of nodes. Other matching methods, such as *maximum weight matching*, are less suitable since they usually focus on maximizing the overall benefit of the mapping of the entire networks.

Take Figure 3 as an illustrative example of different methods. Suppose in Figure 3(a) we are given the similarity scores from the binary classifier for each candidate pair. Figure 3(b) shows that link prediction methods with a fixed threshold (e.g., 0.5) may not be able to predict well, because one customer could be linked with multiple accounts in the social network. In Figure 3(c), *maximum weight matching* methods can find a set of pairs with maximum sum of weights (or similarities), whereas it may not be a good solution for customer identification. Since the similarity score of (u^c_1, u^s_1) is larger than that of (u^c_1, u^s_2), customer u^c_1 is more likely to be the same individual of u^s_1 rather than u^s_2.

Assuming K, the number of customers to be identified, is specified in advance, we propose to find the top K pairs of accounts with the maximum similarity, following the principle of stable matching mentioned above. Figure 3(d) shows an illustrative example of CSI with K=1. Pair (u^c_1, u^s_1) is the top-1 pair that has the maximum similarity score among all candidate pairs. Hence, we would identify u^s_1 as the social network account of customer u^c_1. As a consequence, when K=2 in Figure 3(e), we should ignore the candidate pairs that associate with u^c_1 or u^s_1 due to the one-to-one constraint. Thus, the next pair we would choose is (u^c_2, u^s_2), whose score is the best among the rest pairs. In fact, among all the customers, probably only customer u^c_1 has a user account, u^s_1, in the social network, because the scores of other customers do not indicate that they are similar enough to any users in the social network. Therefore, the result in Figure 3(d) is the most appropriate solution. Nonetheless, we should be able to find the top K-1 pairs before move to the K-th pair, which has lower similarity score than the top K-1 pairs.

```
Algorithm 1: Customer-Social Identification
  Input: a customer-product network $\mathcal{G}^c$, a social
         network $\mathcal{G}^s$, a set of existing identified pairs $\mathcal{A}$,
         and a user-specified value $K$
  Output: a set of predicted pairs $\mathcal{A}'$
  /* first phase*/
  Construct a training set with known labels using $\mathcal{A}$
  For each pair $(u^c_i, u^s_j)$, extract features
  Training classification model $C$ on the training set.
  Perform classification using model $C$ on the test set.
  /* second phase */
  Calibrate the similarity scores of candidate pairs and
  sort them into a max heap $H$ by the scores.
  Initialize all unlabeled $u^c_i$ in $\mathcal{G}^c$ and $u^s_j$ in $\mathcal{G}^s$ as free.
  $\mathcal{A}' = \emptyset$
  while $H \neq \emptyset$ and $|\mathcal{A}'| < K$ do
      Pop the pair $(u^c_i, u^s_j)$ with the max score from $H$
      if $u^c_i$ and $u^s_j$ are both free then
          $\mathcal{A}' = \mathcal{A}' \cup (u^c_i, u^s_j)$
          Set $u^c_i$ and $u^s_j$ as occupied.
```

The proposed CSI method for customer identification is shown in Algorithm 1. In each iteration, we select the pair of accounts (u^c_i, u^s_j) with the maximum similarity score from candidate pairs. If both u^c_i and u^s_j have not yet assigned to any account, we add (u^c_i, u^s_j) to the solution set $\mathcal{A}'$ and set u^c_i and u^s_j as occupied; otherwise if either u^c_i or u^s_j is occupied, we ignore (u^c_i, u^s_j). To facilitate the process of finding the pair with maximum score, we can maintain a max heap instead of a matrix to store the similarity scores of candidate pairs. The algorithm stops when the top K pairs are found, or there are no remaining candidate pairs in the max heap. The matching computed by the CSI method is guaranteed to be a stable matching, according to Theorem 1 in [26]; furthermore, it has the maximum similarity score among all the stable matching of any combination of K pairs across networks, which can be easily proved by mathematical induction. Due to lack of space, we skip all the proofs.

[5] This problem is a variation of maximum weighted stable marriage (or royal couple matching in [26]) problem. The major difference is in that we aim at finding a one-to-one mapping for K nodes, instead of mapping all nodes.

It is worth noting that the selection of the parameter K is a challenging issue for most problems that need to find out the top-K elements. Different approaches are proposed for finding K, such as cross-validation and bootstrapping. In fact, the selection of K can also be implemented in other ways. For example, instead of setting K directly, one can find the top similar pairs until the similarity score of the matching pair is less than a threshold.

4. EXPERIMENTS

In this section, we first introduce the data sets for the experiments, and then present experimental results as well as empirical analysis.

4.1 Data Preparation

We conduct the experiments on the real-world datasets spanning 10 months, as summarized in Table 3. We choose Kickstarter.com, one of the largest sites for crowdfunding[6], as an e-commerce site because the adoption histories of each customer are public available. More importantly, novel and creative crowdfunding projects are notably discussed on Twitter where users are willing to share their interests.

Twitter: We gathered all the tweets regarding Kickstarter from Nov. 2012 to Sep. 2013. For each tweet's author, we queried Twitter API for the metadata about the author as well as the social links of the author. For each project in Kickstarter we consider only the tweets that can link to its webpage. We further filtered out the projects that were seldom discussed (less than 5 tweets) in Twitter. The Twitter dataset after filtering consists of 3,725 projects, 178K users, 5.4 million social links and 385K tweets that construct 234K links between Twitter users and projects.

Kickstarter: We recorded all the projects in Kickstarter launched after Nov. 2012 and completed before Sep. 2013. For each project, we obtained all of its backers, which can be viewed as its customers. For each customer, we crawled his/her user profile and recorded his/her Twitter account, if available. The Kickstarter dataset after filtering consists of 3,725 projects, 545K customers and 868K adoption links between customers and projects. The detailed analysis of these datasets is available in [23].

Data preprocessing: In order to conduct the experiments, we preprocess these raw data to obtain the ground-truth of identified pairs. If a customer, in the Kickstarter dataset, has shown his/her Twitter account in his/her user profile, and the Twitter account also exists in the Twitter dataset, we can safely treat the pair of accounts of the customer as an identified pair. The identified pairs represent the positive instances and they can be used to help construct negative instances of pairs. Due to the one-to-one constraint, we can easily find a negative pair by taking one account from an identified pair and connecting it to any account, in the opposite network, other than the corresponding one. Thus, we can obtain up to 1.3 billion negative pairs.

However, in practice, if an e-commerce company wants to identify one of its customer in a social network, it would probably inquire for the social network accounts whose usernames are similar to the names (i.e., username and full name) of the customer in the company. Consequently, it is critical for the e-commerce company to distinguish the actual one from others with very similar usernames. To simulate the query process, we shall select the negative pairs in which two accounts are likely to have similar usernames. Hence, for each account in the identified pairs, we search the candidate accounts, whose usernames contain a part of the names of the given account, in the opposite network. Then, the candidate accounts are ranked by the Levenshtein edit distance between the candidate usernames and the given customer username. Finally, we sample negative pairs by connecting the given account with up to 100 candidate accounts, other than the corresponding one, with the smallest edit distance. The statistics of the original networks and the sampled networks are presented in Table 3. In the following experiments, we mainly conducted on the sampled networks.

Table 3: Statistics of the datasets.

Kickstarter-Twitter			
	property	original networks	sampled networks
# node	projects	3,725	3,725
	customers	545,638	20,514
	social network users	178,792	43,675
# link	adoption	868,050	39,480
	post	234,550	58,988
	social links	5,467,565	513,651
	identified pairs	1,819	1,819
	negative pairs	1.3 billion	93,436

4.2 Comparative methods

We compare our CSI method with eight baseline methods, which are commonly-used baselines including both supervised and unsupervised link prediction approaches. The compared methods are summarized as follows:

1) **Unsupervised Link Prediction methods:** We compare with a set of unsupervised link prediction methods using the user interest features discussed in Section 3.1: *Common Interests* (**CI**), *Jaccard Coefficient* (**JC**), *Adamic/Adar* index (**AA**), *Resource Allocation* index (**RA**), and *Katz's* index (**Katz**). Following the setting in [20], we test the performance of Katz with three different values of β (i.e., 0.05, 0.005 and 0.0005). Each link predictor outputs a ranked list of candidate pairs in deceasing order of similarity scores. We can evaluate the performance of an unsupervised method based on the ranked list.

2) **Supervised Link Prediction methods:** We test supervised link prediction methods using different types of feature sets separately. As discussed in section 3.1, two feature sets are considered, i.e., **Profile** and **Interest**. We also compare with the combination of both sets of features (**Profile+Interest**). The label predictions of the base classifier are directly used as the final predictions.

3) **Customer-Social Identification (CSI):** The proposed method in this paper. CSI leverages all the extracted features, i.e., Profile+Interest, for training the base classifier. Based on the scores generated by the classifier, CSI takes the top-K maximum similarity and stable matching as the final predictions. In default, K is set as the size of real identified pairs in the testing set. We will analyze the performance of CSI method with K varied in the experiment.

Evaluation Measures. We evaluate the performance of each method in terms of Precision, Recall, F1-score and area under ROC curve (AUC). The first three measures can evaluate the link prediction performances, while AUC evaluates

[6] Crowdfunding – in which people can propose projects and raise funds through collaborative contributions of crowd.

Table 4: Performance comparison of different methods for customer identification. We use different imbalance ratios in both training and test sets. (imbalance ratio = #negative account pairs/#positive account pairs)

		imbalance ratio				
Measure	Methods	10	20	30	40	50
F1-score	Profile	0.672 ± 0.003	0.671 ± 0.003	0.666 ± 0.004	0.665 ± 0.003	0.661 ± 0.004
	Interest	0.836 ± 0.008	0.815 ± 0.011	0.78 ± 0.012	0.763 ± 0.01	0.747 ± 0.006
	Profile+Interest	0.895 ± 0.005	0.875 ± 0.014	0.847 ± 0.017	0.833 ± 0.019	0.803 ± 0.017
	CSI	**0.926± 0.004**	**0.915± 0.003**	**0.898± 0.006**	**0.89± 0.002**	**0.878±0.004**
Precision	Profile	**0.981± 0.001**	**0.977± 0.001**	**0.955± 0.004**	**0.952± 0.002**	**0.935± 0.004**
	Interest	0.944± 0.002	0.932±0.005	0.9±0.006	0.884±0.012	0.868±0.017
	Profile+Interest	0.959±0.004	0.941±0.005	0.925±0.008	0.911±0.006	0.896±0.008
	CSI	0.933± 0.003	0.92±0.003	0.902±0.006	0.894±0.003	0.881±0.003
Recall	Profile	0.511±0.004	0.511±0.004	0.511± 0.004	0.511±0.004	0.511±0.004
	Interest	0.75±0.014	0.725±0.019	0.688±0.021	0.671±0.017	0.656±0.012
	Profile+Interest	0.838±0.009	0.818±0.027	0.782±0.034	0.769±0.035	0.729±0.033
	CSI	**0.92±0.004**	**0.91±0.003**	**0.895±0.007**	**0.887±0.002**	**0.875±0.004**
AUC	Profile	0.791±0.001	0.791±0.001	0.791±0.001	0.792±0.001	0.792±0.001
	Interest	0.933±0.019	0.933±0.019	0.933±0.019	0.924±0.024	0.903±0.018
	CSI	**0.957±0.003**	**0.958±0.004**	**0.958±0.003**	**0.958±0.003**	**0.958±0.003**

Table 5: Performance comparison of different methods for customer identification. We set imbalance ratio $\frac{\#\textbf{negative pairs}}{\#\textbf{positive pairs}} = 50$ and use different number of identified pairs in the training set.

		number of identified pairs in training set (%)				
Measure	Methods	20%	40%	60%	80%	100%
F1-score	Profile	0.342 ± 0.007	0.469 ± 0.159	0.661 ± 0.004	0.661 ± 0.003	0.661 ± 0.004
	Interest	0.486 ± 0.044	0.631 ± 0.026	0.695 ± 0.012	0.720 ± 0.022	0.747 ± 0.006
	Profile+Interest	0.620 ± 0.029	0.753 ± 0.007	0.771 ± 0.014	0.793 ± 0.013	0.803 ± 0.017
	CSI	**0.875± 0.005**	**0.878± 0.004**	**0.877± 0.006**	**0.878± 0.003**	**0.878±0.004**
Precision	Profile	**0.911± 0.006**	**0.924± 0.008**	**0.936± 0.002**	**0.936± 0.004**	**0.935± 0.004**
	Interest	0.900± 0.030	0.893±0.008	0.892±0.003	0.884±0.006	0.868±0.017
	Profile+Interest	0.938±0.016	0.914±0.015	0.900±0.006	0.901±0.005	0.896±0.008
	CSI	0.875± 0.005	0.877±0.004	0.876±0.006	0.878±0.003	0.881±0.003
Recall	Profile	0.211±0.006	0.331±0.150	0.511± 0.004	0.511±0.004	0.511±0.004
	Interest	0.335±0.041	0.489±0.032	0.569±0.016	0.609±0.034	0.656±0.012
	Profile+Interest	0.464±0.033	0.641±0.008	0.674±0.022	0.708±0.021	0.729±0.033
	CSI	**0.875±0.005**	**0.878±0.004**	**0.877±0.006**	**0.879±0.003**	**0.875±0.004**
AUC	Profile	0.773±0.025	0.793±0.002	0.794±0.002	0.793±0.002	0.792±0.001
	Interest	0.903±0.018	0.903±0.018	0.894±0.002	0.904±0.020	0.903±0.018
	CSI	**0.958±0.003**	**0.958±0.003**	**0.958±0.004**	**0.958±0.004**	**0.958±0.003**

the ranking performances. Since unsupervised methods only predict a real-valued score instead of a label prediction for each candidate pair, we only show the AUC performances of unsupervised methods. Moreover, CSI and Profile+Interest share the same set of features and thus they have the same ranking scores generated by the base classifier. Hence, for AUC measure, we use CSI to represent both methods. For fair comparisons, LibSVM [8] of linear kernel with the default parameter is used as the base classifier for all the compared methods. Accuracy is not included in the evaluation measures, since we mainly focus on the real-world imbalanced datasets in which Accuracy is usually meaningless.

Noteworthily, the F1-score and Recall of maximum weight matching (MWM) are consistently lower than 0.1, and the Precision and AUC of MWM are consistently lower than 0.2, which are significantly worse than those of other baseline methods. This is because MWM aims at maximizing the overall benefit of the entire matching instead of the local benefits of individuals, as mentioned in Section 3.2. Since MWM is not suitable for the customer identification problem, MWM is not listed as one of the competitive methods.

4.3 Performance Analysis

We conduct the experiments using 5-fold cross validation: one fold is used as training data, the remaining folds are used as testing data. We report the average results and standard deviations of 5-fold cross validation on the dataset.

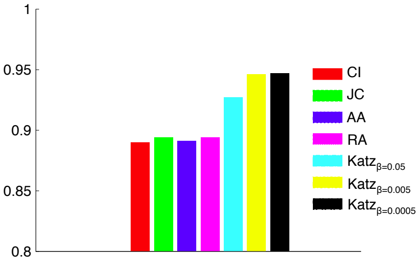


Figure 4: Comparison of customer identification using different features

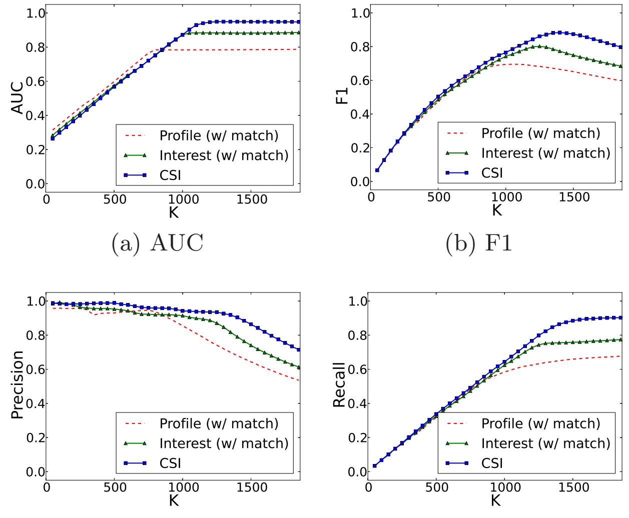
We first investigate the performance of different features in the unsupervised methods. In Figure 4, Katz's methods outperform the methods using other features. It indicates that by exploiting the paths through the interests of friends, we have a better opportunity to identify customers in a social network. Even though the customers are not active in the social network, the common interests with friends may leak the information of the customers and direct us to identify them. However, from the comparison between Katz's methods with different β, which exponentially decreases the weight of longer path, we notice that the importance of friends' interest should not be overrated.

Next, the customer identification problem in real world involves distinguishing the real social network account of a customer from many other similar candidates. If we consider the real pair of accounts as a positive instance and other candidates as negative instances, the number of negative instances usually dominates that of positive instances. In other words, the data instances are usually imbalanced. It is crucial to identify customers in such imbalanced datasets.

Thus, we test the performance of each method with imbalanced datasets. In each round of the cross validation, we sample pairs of accounts as the data samples according to different imbalance ratios. The imbalance ratio is defined as the number of negative pairs divides by the number of positive pairs. Table 4 presents the performance of each method under different imbalance ratios. The best performances on each of the evaluation criteria are listed in bold. The results show that Profile features can be used as the most precise tool to identify some positive pairs but cannot cover most of them. By taking both Profile and Interest features into account, we are able to identify the majority of positive pairs effectively, while only slightly decreasing the precision of identification. The performance can be further improved through the one-to-one matching step in the proposed CSI method. As shown in Table 4, CSI consistently outperforms the other methods in F1-score, Recall and AUC with up to 38%, 80% and 21% improvement, respectively.

Another challenge of customer identification is that, in practice, there are only a small number of identified pairs. Hence, we next study the performance of each method using a small set of identified pairs for training. In each round of cross validation, we randomly sample a percentage of identified pairs from the training fold and use them for training. The results of all compared methods are reported in Table 5. Again, CSI method consistently outperforms other methods in F1-score, Recall and AUC. Especially when only 20% of identified pairs from the training fold are used, the F1-score increases from 0.342 to 0.875 (with 156% improvement) and the Recall increases from 0.211 to 0.875 (with 315% improvement). We also notice that the performance of CSI method is quite stable with the change of the number of training samples. This is because CSI method is designed to find the best stable matching all the time. Lack of training samples only affect the accuracy of similarity scores, while it probably would not change the preference of each account.

Finally, we investigate the performance of each method with K varied, where K is the number of pairs we should find in a one-to-one matching. In our experiments, K is set as 1466, which is the size of real pairs in our testing set, in default. Since the predictions of classifications cannot be directly compared, we won't be able to find the top-K pairs using the above baseline methods. Thus, in this experiment, we compare the performance of the CSI methods using different sets of features. We denote the CSI method using only Profile feature set as "Profile (w/ match)", and we denote that using only Interest feature set as "Interest (w/ match)". Figure 5 shows that CSI method incorporating the more features can achieve the better performance. Besides, the CSI method using only the Interest feature set performs better than that using only the Profile feature set. More importantly, our proposed CSI method achieves the best performance when K is around 1466, the actual size of pairs to be identified. This indicates that CSI method can effectively find the top-K pairs that are most likely to be the real pairs before it moves to pick the less possible pairs.



(a) AUC

(b) F1

(c) Precision

(d) Recall

Figure 5: Comparison between CSI and baselines with K varied

5. RELATED WORK

Due to the emergence of online social network services, social network analysis have been intensively studied in recent years [24, 18, 21, 12]. One active research topic is to predict unknown link in social network. Liben-Nowell and Kleinberg [20] developed unsupervised link prediction methods based upon several topological features. These proposed features can be further used to train a binary classification model for link prediction. There are many other recent efforts on link prediction problem in social networks. For example, in [2], Backstrom et al. proposed a supervised random walk algorithm to estimate the strength of link in social networks. Lichtenwalter et. al. [21] have a detailed discussion over different challenges of link prediction problem. Kong et al. [18] formulated the problem of connecting accounts across social networks as a anchor link prediction task, w.r.t one-to-one constraint across social networks. They leverage the heterogeneous features, such as social, spatial and temporal information, to help predict the anchor links.

Recently, user identification across multiple social networks has attracted many attentions [31, 22, 27, 25]. Zafarani et al. [31] observed that individuals often exhibit consistent behavioral patterns across networks when selecting usernames. Based on the observation, they proposed a behavior model to determine whether two usernames are belong to the same individual. In [27], Raad et al. addressed the problem of matching user profiles for inter-social networks. [25] analyzed users' online digital footprints and applied context specific techniques to measure the similarity of accounts across networks. These studies indicate that username is one of the most discriminative features for disambiguating user profiles. However, customer identification has some unique properties that make it different to the previous works. First, it requires to predict links across networks with completely different schema (i.e., bipartite network vs. general heterogeneous network). Second, since most networks are partially aligned, we should identify the most similar pairs instead of mapping the entire networks. Due to these issues, previous approaches may not be directly applicable to customer identification.

6. CONCLUSION

In this paper, we have described and studied the problem of customer identification in social networks. Different from previous works in link prediction and network alignment, customer identification requires to predict links between accounts across partially aligned networks with completely different schema. We have proposed to extract two types of features, user profile features and user interest features, that can be used to compute the similarity scores of pairs across such networks. By finding the top-K maximum similar and stable matching, our proposed approach CSI (Customer-Social Identification), can effectively connect customers with their corresponding social network accounts. Extensively experiments have demonstrated that our CSI method consistently outperforms other commonly-used baselines on customer identification.

In summary, this work provides a promising step towards incorporating existing online social networks for e-commence. By leveraging the rich information from social networks, many potential applications could be developed. Examples of applications include personalized product recommendation using social correlations, spam detection by identifying true customers, and maximization of product adoption and profits over social networks.

Acknowledgment

This work is supported in part by NSF through grants CNS-1115234, and OISE-1129076, US Department of Army through grant W911NF-12-1-0066, and the Pinnacle Lab at Singapore Management University.

7. REFERENCES

[1] L. A. Adamic and E. Adar. Friends and neighbors on the web. *Social Networks*, 25(3):211–230, 2003.

[2] L. Backstrom and J. Leskovec. Supervised random walks: predicting and recommending links in social networks. In *WSDM*, pages 635–644, 2011.

[3] M. Bayati, D. F. Gleich, A. Saberi, and Y. Wang. Message-passing algorithms for sparse network alignment. *TKDD*, 7(1):3, 2013.

[4] N. Benchettara, R. Kanawati, and C. Rouveirol. Supervised machine learning applied to link prediction in bipartite social networks. In *ASONAM*, pages 326–330, 2010.

[5] S. Bhagat, A. Goyal, and L. V. S. Lakshmanan. Maximizing product adoption in social networks. In *WSDM*, pages 603–612, 2012.

[6] R. Bhatt, V. Chaoji, and R. Parekh. Predicting product adoption in large-scale social networks. In *CIKM*, pages 1039–1048, 2010.

[7] I. Bhattacharya and L. Getoor. Collective entity resolution in relational data. *TKDD*, 1(1), 2007.

[8] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM TIST*, 2:27:1–27:27, 2011.

[9] F. C. T. Chua, H. W. Lauw, and E.-P. Lim. Generative models for item adoptions using social correlation. *TKDE*, 25(9):2036–2048, 2013.

[10] D. J. Crandall, D. Cosley, D. P. Huttenlocher, J. M. Kleinberg, and S. Suri. Feedback effects between similarity and social influence in online communities. In *KDD*, pages 160–168, 2008.

[11] L. E. Dubins and D. A. Freedman. Machiavelli and the Gale-Shapley algorithm. *American Mathematical Monthly*, 88(7):485–494, 1981.

[12] L. Getoor and C. P. Diehl. Link mining: a survey. *SIGKDD Explorations*, 7(2):3–12, 2005.

[13] S. Guo, M. Wang, and J. Leskovec. The role of social networks in online shopping: information passing, price of trust, and consumer choice. In *ACM EC*, pages 157–166, 2011.

[14] S. Hill, F. Provost, and C. Volinsky. Network-based marketing: Identifying likely adopters via consumer networks. *Statistical Science*, 22(2):256–275, 2006.

[15] M. Hu and B. Liu. Mining and summarizing customer reviews. In *KDD*, pages 168–177, 2004.

[16] M. Jiang, P. Cui, R. Liu, Q. Yang, F. Wang, W. Zhu, and S. Yang. Social contextual recommendation. In *CIKM*, pages 45–54, 2012.

[17] L. Katz and L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, Mar. 1953.

[18] X. Kong, J. Zhang, and P. S. Yu. Inferring anchor links across multiple heterogeneous social networks. In *CIKM*, pages 179–188, 2013.

[19] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady.*, 10(8):707–710, Feb. 1966.

[20] D. Liben-Nowell and J. M. Kleinberg. The link prediction problem for social networks. In *CIKM*, pages 556–559, 2003.

[21] R. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In *KDD*, pages 243–252, 2010.

[22] J. Liu, F. Zhang, X. Song, Y.-I. Song, C.-Y. Lin, and H.-W. Hon. What's in a name?: an unsupervised approach to link users across communities. In *WSDM*, pages 495–504, 2013.

[23] C.-T. Lu, S. Xie, X. Kong, and P. S. Yu. Inferring the impacts of social media on crowdfunding. In *WSDM*, pages 573–582, 2014.

[24] Z. Lu, B. Savas, W. Tang, and I. S. Dhillon. Supervised link prediction using multiple sources. In *ICDM*, pages 923–928, 2010.

[25] A. Malhotra, L. C. Totti, W. M. Jr., P. Kumaraguru, and V. Almeida. Studying user footprints in different online social networks. In *ASONAM*, pages 1065–1070, 2012.

[26] A. Marie and A. Gal. On the stable marriage of maximum weight royal couples. In *IIWeb*, 2007.

[27] E. Raad, R. Chbeir, and A. Dipanda. User profile matching in social networks. In *NBiS*, pages 297–304, 2010.

[28] M. Tsytsarau, S. Amer-Yahia, and T. Palpanas. Efficient sentiment correlation for large-scale demographics. In *SIGMOD Conference*, pages 253–264, 2013.

[29] J. Yan, A. Blackwell, R. Anderson, and A. Grant. The memorability and security of passwords - some empirical results. Technical report, 2000.

[30] B. Zadrozny and C. Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *KDD*, pages 694–699, 2002.

[31] R. Zafarani and H. Liu. Connecting users across social media sites: a behavioral-modeling approach. In *KDD*, pages 41–49, 2013.

[32] T. Zhou, L. Lü, and Y.-C. Zhang. Predicting missing links via local information. *The European Physical Journal B - Condensed Matter and Complex Systems*, 71(4):623–630, Oct. 2009.