Prediction of purchase behaviors across heterogeneous social networks

Yuanzhuo Wang[1] · Jingyuan Li[1] · Qiang Liu[1,2] · Yan Ren[3]

Published online: 12 August 2015

Abstract Due to the development of web services, many social network sites, as well as online shopping sites have been booming in the past decade, where it is a common phenomenon that people are likely to use multiple services at the same time. On the one hand, previous research findings indicate the data sparsity issues of online shopping accounts, which is caused by the heavy-tailed distribution of user information. On the other hand, in social network sites, the personal information and the corresponding statuses of an account are abundant, and their genuineness is guaranteed either by the service provider, or by the willingness of the account owner to connect to his or her friends in reality. Making use of the correlation between accounts of a same individual is a crucial prerequisite for many interesting cross network applications, such as improving the recommendation performance of the online shopping sites using extra information from social network services. In this paper, we firstly propose a game-theoretic method to identify correlation accounts of individuals between social network sites and online shopping sites with stable matching model, incorporating account profiles as well as historical behaviors. Using the above account relationships, we then put forward a predicting method that combines heterogeneous social network information and online shopping information, to predict the purchasing behaviors. The results show that our method identifies up to 70 % of the correlation accounts between Facebook and eBay, one of the most popular social network sites and online shopping

✉ Yuanzhuo Wang
wangyuanzhuo@ict.ac.cn

[1] CAS Key Lab of Network Data Sci & Tech, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China

[2] University of Chinese Academy of Sciences, Beijing, China

[3] National Computer Network Emergency Response Technical Team Coordination Center of China, Beijing, China

sites in the world, respectively. The experimental results also show that using the correlation account sets, the accuracy of our purchase predicting method outperforms the state-of-the-art methods by 5 %.

Keywords Prediction of purchase behaviors · Account matching · Heterogeneous social networks

1 Introduction

With the developments of WWW, many user-centric web services are developed so well that they have become the leading power of the world for the last decade [1], where social network services and online shopping services are two shining representatives, owning billions of users all over the planet. Social network services, such as Facebook or Google+, strengthen self-status updating and friend communication, while online shopping services, e.g., eBay or Taobao, concern commodities exchange and purchase. The two seemingly isolated web services have been connected by people: a great percentage of people have accounts of both social network and online shopping services. Matching a person's accounts in different services would be a crucial prerequisite for calculating many useful outcomes [2]. For example, it can be very profitable if we collect a person's profiles and actions in social networks to categorize or even predict the purchase behaviors of the same person in online shopping sites [3]. Conversely, the purchasing data may be used by social network service providers to recommend friends and cast customized advertisements with a higher precision [4].

However, account matching is not an easy task because of the lack of information across the networks [5]. A bunch of solutions have been proposed to solve the problem, where most of them focus on the situation where account matching occurs between two homogeneous networks [2,4], such as the matching between Facebook and Twitter. Although the domains of the two comparing networks are not completely identical, most of them concern similar characteristics. When applied to heterogeneous services, like Facebook and eBay, the gap between the domains defeats most of the well-known account matching algorithms.

The user matching task across multiple heterogeneous networks is very challenging due to the specific characteristics of the task. On the one hand, it is difficult to extract key features from one type of web services, and conceptualize them to describe the account of a completely different type of services with a proper explanation. On the other hand, the actions of an account in one service may not comply with those actions in another service. To overcome these challenges, it is necessary to generalize the two different categories of services, and use a more commonly applicable model to describe the matching issue.

In this paper, we introduce a game model called stable matching to solve the account matching problem. Stable matching is a quite generalized methodology for solving the match issues [6], such as the perfect matches of a list of men and a list of women, or a list of students and a list of universities. The key point is that the method can abstract the domains or characteristics of an account into certain utility functions, which might be a way to connect two otherwise very different types of objects.

Moreover, we further study on applying the account matching results into purchase processes, so as to give an example of the role of our matching results in prediction across heterogeneous networks. We use the personal information and statuses in the social network site [7] of a matched account pair, and the corresponding purchasing histories, to predict a possible likelihood of the future purchasing behaviors [8].

The main contributions are as follows:

- We propose a series of strategies to calculate account similarity based on account demographic difference and interest vector, and then use the game version of stable matching to solve account matching in heterogeneous networks.
- Based on the matching results, we propose a predicting method that combines heterogeneous social network information and online shopping information, to predict the purchasing behaviors.

We conduct a bunch of experiments through real-world datasets collected from the world's number one social network sites Facebook, and one of the largest online shopping sites eBay, and the results have proved the correctness and effectiveness of our methods.

The rest of the paper is organized as follows. In Sect. 2, we review some related work in account matching. In Sect. 3, we focus on extending the stable matching model to solve the account matching issue, and presenting the conversion procedure from user behaviors to interest vector. We then describe our purchase predicting method in Sect. 4. Experiments and results are shown and discussed in Sect. 5, and Sect. 6 concludes the paper.

2 Related work

Predicting users' likes have received increasing attention and have been thoroughly studied over the past decades. The methodologies are twofold: collaborative filtering and content-based methods. Collaborative filtering [9] assumes that users who carry similar characteristics tend to like similar products, including user-based and item-based approaches. User-based collaborative filtering [10] predicts items to a target user using collected information from similar users, while users are typically represented in a vector space which summarizes their characteristics. Similarly, item-based collaborative filtering methods [11] take advantage of rating information of similar items which are reviewed by the target user in the past. In contrast to collaborative filtering, content-based methods often utilize the vast overload of information in the web site [12], such as product reviews, customer opinions, and social media to directly make product recommendation.

Although those approaches are studied extensively, data sparsity makes many well-known prediction approaches perform poorly, such as in a cold-start situation [3,13]. Researchers proposed hybrid approaches to incorporate both user-item rating dataset and other contextual information in different scenarios, including social network information and time information [14]. For instance, social trust or friend aware recommender approaches model trustworthiness or similarities of users. Lin [13] combined the feature of Twitter followers, and generated a much more accurate estimation of how likely a target user would purchase an App. Zhang [15] showed that there are significant correlations between social network information and online purchases, and presented a system that uses Facebook likes for solving the task of products' recommendation. However, the above research findings are all based on a strong correlation assumption between user accounts, and could not fully take advantage of the rich features in user social profiles, like descriptions, which are poorly structured and difficult for machine to understand.

Account matching across heterogeneous networks has received increasing attention and numerous solutions have been proposed. The methodologies are threefold: unsupervised matching, supervised matching and game-theoretical methods. Unsupervised matching assumes that individuals prefer to be friends with the ones having similar profile attributes and friend networks, which can be further categorized into two categories: semantic-based and structure-based approaches. Semantic-based unsupervised matching judges the closeness of accounts using the same or similar profile attributes of the two accounts. Buccafurri [16] proved that two nodes are similar and therefore likely to refer to the same individual, if they have similar usernames. However, the target online services may fall into radically distinct categories with very different profile attributes, where the semantic-based approaches may not be applicable at all. Structure-based methods utilize the individuals' friend network or any other connective attributes to judge the closeness of the accounts. Bartunov [17] demonstrated the importance of social links for identity resolution task and showed that the solution significantly outperforms the attribute-based approaches. Nevertheless, connective attributes are commonly important in social networks, which may not be as equal important, or even invincible in some services, like online shopping.

To overcome the above shortages, supervised account matching methods have been proposed in recent years, where the features used in unsupervised approaches are directly applied to train classification models. Peled [18] presented a supervised learning method to match user profiles, and then proved its good performance in matching user profiles, though incomplete records with missing data could significantly increase the error rate of the comparison algorithm. Xu [19] showed that combing social features with personal features could improve the performance of criminal identity matching. Enlightened by Xu's work, we combine profile attributes and historical behaviors together in our account matching algorithm. A very different type of approach or account matching goes to game theory. Shehab [20] proposed a method leveraging the game appeal and social community to generate the profile mappings, in which procedure the game was modeled using incomplete information, and a proof of sequential equilibrium was given. Kong [2] formulated the inference problem for anchor links into a stable matching problem between the two sets of accounts in two different networks, who claimed that their methods can effectively predict the links between accounts. We applied stable matching model in our account matching procedure. A typical scene of stable matching is to assign students to colleges, where students and colleges have quite different attributes. This matching characteristic of heterogeneous objects might be the key to solve our account matching problem.

Motivated by previous work, we firstly study on the account matching issue based on stable matching model, and then try to predict purchase behaviors in online shopping sites utilizing the social characteristics in social network sites.

3 Account matching across heterogeneous networks

Stable matching has been the object of intensive study by both computer scientists and economists. The classical example of stable matching is to match between a set of men and a set of women so that they can be married happily. In the classical version of stable matching, it requires that each node has a strict ordering of preference list over the opposite nodes. In addition, for each pair of agents, m preferring w is not necessarily equal to w preferring m. However, in account matching between, say social networks and online shopping, the above two conditions do not always apply, especially when the preference of nodes is calculated with user profiles and historical behaviors.

3.1 Problem definition

Suppose we are given a social network data G^S and an online shopping data G^E, and suppose the social network G^S has comprehensively detailed user profiles and sufficient real-time post-messages while the online shopping data G^E are without any information but user historical behaviors. The social network $G^S = (U, F^S, M^S)$ contains account profiles and post-messages, where $U = \{u_1, u_2, \ldots, u_n\}$ is the set of user accounts, $F^S = \{f_1, f_2, \ldots, f_k\}$ is the profile attributes, e.g., username, nickname, and M^S represents the text contents of the statuses published in the social network. $G^E = (U, F^E, H^E)$ denotes the same user accounts U, but with less profile attributes F^E. We define similarity S as the closeness of two accounts between social network and online shopping services, which is the undirected replacement of the otherwise directed preferences in our extended stable matching model.

Definition 1 (*Account matching*) Suppose there are a social network service G^S and an online shopping service G^E. Account matching is a one-to-one mapping between user accounts in G^S and G^E.

3.2 Features extraction of social network account

When someone joins a social network site, he or she may fill in some personal information, like name, gender and birthday, or even a self-description. The person can then update his or her statuses by posting short texts or sharing photos. We are to use the above profile attributes and historical behavioral messages for account matching.

Profile attributes A large amount of work has studied the demographic distinctions in personalized services. Raad [21] used most of the profile attributes, like username, nickname, mailbox, image, etc., and gave different importance to the attributes. Lofciu [22] combined user tags in user profiles to measure the distance between user profiles for identification.

Historical messages When people register for a social network site, they usually publish messages of their own, or propagate their friends messages. Numerous studies have proved that the message of an account can be of great value to the process of personalizing the features of the corresponding person. Elnaz [23] showed that the presence of social components has a positive impact in increasing the accuracy of hybrid recommendation. However, the messages published are usually short. To normalize the messages into a fixed-length feature vector, we use methods as follows.

- Corpus-based: computing the interest vector using large corpora only (without external knowledge resources). The method is based on vector-space model, such as Latent Dirichlet Allocation, or Latent Semantic Analysis.
- Knowledge-based: computing the interest vector with the use of predefined (or external) knowledge resources (taxonomies, ontology, etc) such as WordNet and Wikipedia. The meaning of the interest vector can be reinforced by the related concepts of the knowledge base.

For the processing of historical messages in social networks, we apply the aforementioned corpus-based and knowledge-based methods, respectively. The two different computing methods have a same denotation $F_u^T = \{c_i : w_i^T\}$, where c_i is a feature: a keyword for corpus-based learning and a topic for a knowledge-based learning. w_i is the weight of the feature, which can be the number of a word that appears in the text for corpus-based learning, or the number of a concept that appears in the text.

3.3 Features extraction of online shopping account

When compared to social network account, user profile attributes in online shopping sites are sparse. However, some attributes, like nickname, are quite important in account matching. Studies have shown that individuals tend to use the same username, or a similar one in different online services. Peled [18] believed that name-based features are the most important ones in entity matching. Therefore, we take online shopping account profile attributes into consideration.

Contrasting to the sparsity of profile attributes, an individual's historical behavior in online shopping site is abundant. Liu [24] proposed a recommendation framework to predict users interest using past click history, which improved the quality of news recommendations. We convert user behaviors into interest vector, where the target online shopping service providers' goods categories are inherited. For example, eBay divides its products into 36 categories. We collect users historical behaviors, and classify the related product into categories, which is represented as an interest vector.

To be specific, we turn historical behaviors into categories that is denoted as $F_u^B = \{b_i : w_i^B\}$, where b_i is a category and w_i describes the likelihood of the user for that category.

3.4 Similarity between accounts

We apply three methods to calculate the similarity between accounts.

Words-distance Individuals tend to use the same, or similar usernames. Therefore, we use Levenshtein distance to describe such characteristics, which is the minimum number of single character editions (insertion, deletion and substitution). The similarity of two usernames u_i and u_j is computed as

$$W_D = 1 - \frac{d_{\text{lev}}(u_i, u_j)}{\max(\text{len}(u_i), \text{len}(u_j))}, \tag{1}$$

where $d_{\text{lev}}(u_i, u_j)$ is the Levenshtein distance of u_i and u_j, $\text{len}(u_i)$ is the number of characters of u_i.

One-zero type Attributes like gender, mailbox or location are very informative, because they could identify a person uniquely. If two profiles are associated with the same email address, it is highly likely that the two accounts refer to a same individual. To compare the values of those attributes, we need to determine whether the attributes of a profile are identical to the other profile. In this case, the similarity is assigned 1 for a positive answer, and 0 for a negative one.

Interest-vector distance We conceptualize user historical behavior into interest vector. Due to the characteristics of different online services, individuals' interests maybe presented very differently from each other, therefore it is unpractical to calculate the similarity by methods like Jaccard distance. We train relationship between user interests, and based on the relationship to predict users' likes.

3.5 Account matching algorithm

We apply the three aforementioned strategies to calculate the account similarity; the result obeys the one-to-one mapping, as shown in Algorithm 1.

In each iteration of Algorithm 1, we firstly randomly select a free node n_i^A in A from the source network. Then we get the most preferred agent n_j^B by n_i^A in its preference list $S_{n_i^A}$. If n_j^B's most preferred agent in its preference list is n_i^A, then we believe the pair $\{n_i^A, n_j^B\}$ is a stable pair. Otherwise, there exists a better stable pair for $n_{j'}^B$. If the pair $\{n_i^A, n_j^B\}$ is stable, n_i^A and n_j^B are already occupied by each other. Therefore, for each node $n_{j'}^B$ which prefers n_i^A, and for each node $n_{i'}^A$ which prefers n_j^B, there is no chance to be a stable matching.

Algorithm 1 Account stable matching

```
Require: two disjoint agents $N^A$ and $N^B$
Require: similarity between agents $S$
Ensure: stable matching agent pairs
pairs = ∅
while exists free $n_i^A$ in $N^A$ and $n_j^B$ in $N^B$ do
  if $n_i^A$'s most preference $n_j^B$ and $n_j^B$'s most preference is $n_i^A$ then
    remove $n_k^B$'s preference to $n_i^A$ in $B$
    remove $n_k^A$'s preference to $n_j^B$ in $A$
  end if
  put $\{n_i^A, n_j^B\}$ into pairs
end while
```

Algorithm 1 has the following three properties.

Property 1 (Convergence) *At least one pair of nodes will be calculated in each step, therefore the algorithm terminates within finite steps.*

Proof Suppose there is no pair of nodes chosen in some step, it means that when node n_i^A proposes to n_j^B, node n_j^Bs top preference is not n_i^A, but $n_{i'}^A$, therefore $S(n_j^B n_{i'}^A) \succ S(n_j^B n_i^A)$. However, we know that $S(n_j^B n_{i'}^A) = S(n_{i'}^A n_j^B)$, and $S(n_j^B n_i^A) = S(n_i^A n_j^B)$, and so $S(n_{i'}^A n_j^B) \succ S(n_i^A n_j^B)$, which contradicts the fact that n_i^A is the top most node of n_j^Bs list. □

Property 2 (Stableness) *The algorithm terminates in a stable matching.*

Proof Suppose there exists an unstable result. Then there exists a blocking pair $\{n_i^A, n_j^B\}$ with n_i^A matching to $n_{j'}^B$, and n_j^B matching to $n_{i'}^A$. Since $\{n_i^A, n_j^B\}$ is blocking and $n_j^B \succ_{n_i^A} n_{j'}^B$, in Algorithm 1, n_i^A would have proposed to n_j^B before $n_{j'}^B$. Since n_i^A is not matched with n_j^B by the algorithm, it must be because n_j^B receives a proposal from a node $n_{i'}^A$ that is higher than n_i^A. Since the n_j^B is matched to $n_i^{\prime A}$, it follows $n_{i'}^A \succ_{n_j^B} n_i^A$. This contradicts the fact that $\{n_i^A, n_j^B\}$ is a blocking pair. □

Property 3 (Uniqueness) *No matter which side proposes, the result of account matching is unique.*

Proof In each iteration, no matter which side proposes, the algorithm chooses the node with the top preference. Suppose that there exist a pair of nodes $\{n_i^A, n_j^B\}$ when proposed from A, they have not been selected when proposed from B. It means than n_j^B's most preference node is not n_i^A, which contradicts to the fact that n_j^B's most preference node is n_i^A when $\{n_i^A, n_j^B\}$ is proposed from A. □

4 Behavior prediction across heterogeneous networks

In Sect. 3, we gave a game-theoretic matching model to connect two heterogeneous networks with accounts. In this section, we will use the matching results and the historical purchase data to build a predicting model for future purchase behaviors.

4.1 Problem definition

Suppose there is an online shopping network, $U = \{u_1, u_2, \ldots, u_n\}$ denotes the users (or accounts), $C = \{c_1, c_2, \ldots, c_m\}$ denotes the categories of products. The problem is defined as to find a ranking of all categories for target users, according to the rating score $P(u, c)$, which denotes user u's preference to category c.

The historical purchase behavior H_u, although very sparse, has to be considered in purchase prediction, because it is a 100 % ground truth of the user. We could use this information as a predicting dimension, which is depicted as $P_H(u, c)$. Additionally, we could use user profiles D_u and posted messages M_u as predicting aspects $P_D(u, c)$ and $P_M(u, c)$, respectively. By doing so, we could enrich the ingredients of the predicting recipe from social network sites.

We consider all the aforementioned information, and merge it into one prediction outcomes by giving a group of weight parameters α, β, γ to each predicting dimension as shown in Eq. (2). Through the following subsections, we will present detailed descriptions to the three predicting dimensions, and put forward the method to calculate the weight parameters.

$$P(u, c) = \alpha P_H(u, c) + \beta P_D(u, c) + \gamma P_M(u, c) \tag{2}$$

4.2 Prediction with purchase history and product categories

Users' purchase history is one of the key issues we should take into consideration [25], because it is the first-hand truth of user likes, even though data sparsity may prevent the importance of its predicting results. Moreover, purchase behaviors have the characteristic of time sequencing, therefore, recent purchase behaviors have more guiding values than aging ones.

We split the historical purchase behaviors of a user u into periods of length t, where we define N_u^t as the total number of purchases of a user, $N_u^t(c)$ as the number of purchases of category c, and N^t as the total number of purchase behaviors. Then u's preference to c_i can be denoted as $p_u^t(c_i|L) = (N_u^t(c_i))/(N_u^t)$, purchase ratio within time t is $p_u^t(L) = (N_u^t)/N^t$, purchase ratio of c_i within time t is $p^t(c_i) = (N_u^t)/N^t$. According to Bayes model, we can denote the preference of user u to category c_i as

$$p_u^t(L|c_i) = \frac{p_u^t(c_i|L) \cdot p_u^t(L)}{p^t(c_i)} \tag{3}$$

To strengthen the timeliness of the user preference, we apply Gaussian equation $\mathcal{N}(\mu, \delta^2)$ to describe the weight of preference according to t, where μ is the target time, δ is to describe the smoothness of Gaussian equation. The predicting function of $P_H(u, c_i)$ can therefore be expressed as Eq. (4).

$$P_H(u, c_i) = \frac{\sum \mathcal{N}_t(\mu, \delta^2) \cdot N_u^t \cdot p_u^t(L|c_i)}{\sum \mathcal{N}_t(\mu, \delta^2) \cdot N_u^t} \tag{4}$$

4.3 Prediction with social network information

Based on the matched accounts, or the strong relationship between social network sites and online shopping sites, we can use both user profiles and the published statuses to predict the purchase behaviors of the future.

Social network demographical characteristic When someone signs up an account in a social network site, he or she would probably provide personal information to the service provider, e.g., name, gender, age, or even religious belief. Authentic personal information helps the new user to find his or her friends in reality, and makes it easy for other users to find him or her with similar hobbies or interests [26]. For gender, we merely match users with the same gender. Let $B(u_j)$ be user u_j's match result, where 1 represents the same gender, 0 otherwise. We also assume that closer users are likely to have closer preferences with high probability. For age, we use Gaussian equation to calculate the distance between the target user and the training users, as shown in Eq. (6), and sum the results together to assign to the target user. Similarly, we could calculate the distance between locations. Finally, we sum all three of the characteristics together with weights, as depicted in Eq. (8).

In more details, we firstly calculate the relation between social networks and online shopping sites based on the training set. The simple version is to count the shopping behaviors for each social network feature $L(c, b_j) = \sum_{k=1}^{n} w_k^{b_j}$, where c denotes a social network feature, b_j is a kind of behavior in online shopping sites, n is the number of users in the training set, and $w_k^{b_j}$ is the weight of user k's behavior. The advanced version is to add relation into the learning of social network features regarding the likes of online shopping sites $B(u) = \sum_i^k w_u^{ci} L(c, b)$, where k denotes the number of likes of the user, w_u^{ci} denotes the weight of c_i for user u, $L(c, b)$ is the correlation between likes and behaviors. In Eq. (5), n denotes the number of users in the training set, $B(u_j, c_i)$ shows if u_j has feature c_i (1 if yes, 0 otherwise). $w(c_i)$ denotes the weight of social network feature c_i, M is the number of features u_j has, and $w(c_k)$ is the weight of social network feature c_k of user u_j. In Eq. (7), θ_i denotes the weight. Equation (8) is the fusion of the three aspects $p_G(u, c_i)$, $p_A(u, c_i)$ and $p_L(u, c_i)$, where $p_L(u, c_i)$ is the prediction according to locations.

$$p_G(u, c_i) = \sum_j^n B(u_j, c_i) \frac{w(c_i)}{\sum_k^M w(c_k)} \tag{5}$$

$$D(u_i, u_j) = \mathcal{N}(\mu, \delta^2) \tag{6}$$

$$p_A(u, c_i) = \sum_j^n D(u, u_j) \frac{w(c_i)}{\sum_k^M w(c_k)} \tag{7}$$

$$p_D(u, c_i) = \theta_1 p_G(u, c_i) + \theta_2 p_A(u, c_i) + \theta_3 p_L(u, c_i) \tag{8}$$

Social network user statuses Social network is with billions of user interactions, most of which are short text messages [24]. We could utilize the messages to build a model to describe user preferences, by which we can recommend friends of the user. We use an open source knowledge base Freebase to understand the short messages, and to learn the preference model. For semantic recommendation strategies, methods based on matrix factorization models are the state-of-art approaches in recommender system.

Matrix Factorization (or MF) [27] is one of the common methods for model-based recommendation. MF has been proposed to perform predictions for a single user-item rating matrix. In MF, each user and each item is associated with a K dimensional latent factor vector: the latent factor of user u is denoted as U_u and is stored as the uth row of user factor matrix U. The latent factor of item i is denoted as V_i and stored as the ith row of item factor matrix V. To learn the latent factors of users and items, [28] employs probabilistic matrix factorization to factor the user-item matrix into the product of user and item latent factors. The conditional probability of the observed ratings is defined as the following equation:

$$P_M(u,c) = p(RU, V, \delta^2) = \prod_{u=1}^{N}\prod_{i=1}^{M}[\mathcal{N}_t(R_{u,i}|U_u^T V_i, \delta_r^2)]^{I_{u,i}^R} \tag{9}$$

where $\mathcal{N}_t(x|\mu, \delta^2)$ is the normal distribution with mean μ and variance δ^2, and $I_{u,i}^R$ is the indicator function that is equal to 1 if u has been rated i, and is equal to 0 otherwise.

4.4 The combination of predicting dimensions

We apply the aforementioned three predicting dimensions, respectively, and then give each of them a weight to form a mixed result as shown in Eq. (10). We name our method as *F*usion of *H*eterogeneous *S*ocial Network Information and *O*nline Shopping for User Preference *P*rediction (FHSOP)

$$P(u,c) = \alpha P_H(u,c) + \beta P_D(u,c) + \gamma P_M(u,c) \tag{10}$$

The parameters can be learned by maximizing the above objective function using the stochastic gradient descent (SGD) algorithm. SGD has a fast speed to convergence and a high scalability to large-scale data sets. The main process of SGD is to randomly scan all training instances and iteratively update parameters. We have the gradients as follows:

$$J(\alpha,\beta,\gamma) = \frac{1}{2}\sum_{i=1}^{m}(P(u,c) - y(u,c))^2 \tag{11}$$

$$\frac{\partial J(\alpha,\beta,\gamma)}{\partial\alpha} = \sum_{i=1}^{m}(Pu, c - y(u,c)) \cdot P_H(u,c) \tag{12}$$

$$\frac{\partial J(\alpha,\beta,\gamma)}{\partial\beta} = \sum_{i=1}^{m}(Pu, c - y(u,c)) \cdot P_D(u,c) \tag{13}$$

$$\frac{\partial J(\alpha,\beta,\gamma)}{\partial\gamma} = \sum_{i=1}^{m}(Pu, c - y(u,c)) \cdot P_M(u,c) \tag{14}$$

$$\alpha_{k+1} = \alpha_k + \varepsilon \cdot \frac{\partial J(\alpha,\beta,\gamma)}{\partial\alpha} \tag{15}$$

$$\beta_{k+1} = \beta_k + \varepsilon \cdot \frac{\partial J(\alpha,\beta,\gamma)}{\partial\beta} \tag{16}$$

$$\gamma_{k+1} = \gamma_k + \varepsilon \cdot \frac{\partial J(\alpha,\beta,\gamma)}{\partial\gamma} \tag{17}$$

where ε is the predefined step size.

5 Experiments

5.1 Datasets

To evaluate the performance of account matching, we collected data from Facebook, the largest social network in the world, and eBay, the largest e-commerce service in the world. Facebook data are with abundant user profiles and statuses, while eBay data have sufficient historical purchasing information.

We collected 507 accounts from Facebook. where 58.06 % of the accounts belonged to male individuals, and 41.94 % belonged to female individuals, which is consistent to the gender distribution published by Facebook. We gathered 239, 772 post-messages from Jan 2014 to Jun 2014, and observed that people's interests changed over time.

We also collected eBay accounts which had at least one purchase behavior between Jan and Jun 2014. The account set had 31, 865 purchase behaviors in total. Figure 1 displays the power-law distribution of user purchased products. Figure 2 shows the number of purchases according to product categories in eBay.

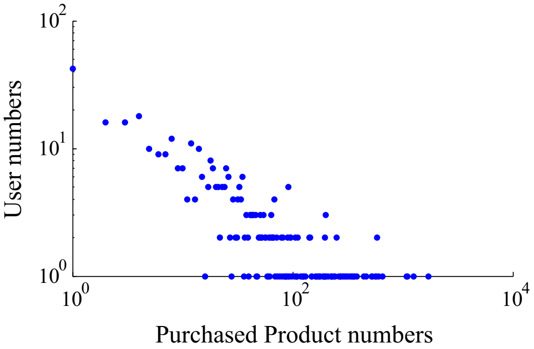

Fig. 1 Power distribution of users' likes

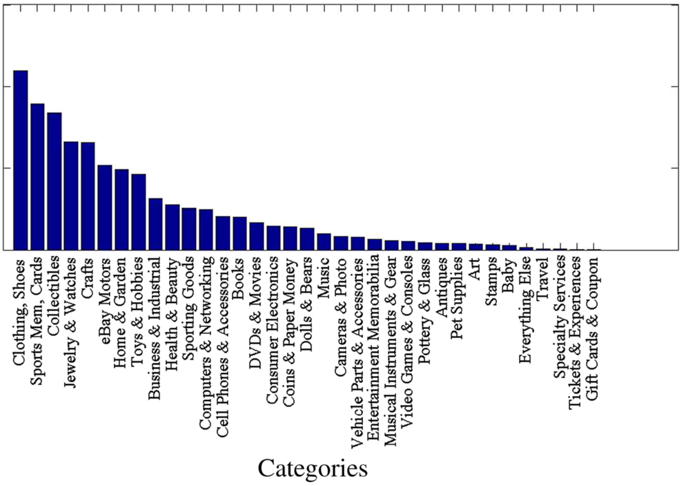

Fig. 2 Distribution of purchase categories

5.2 Experiments on account matching

5.2.1 Methodology and comparison

Methodology To evaluate the efficiency of the proposed approach, we compared our method with several baseline methods. The compared methods are summarized as follows:

(1) Unsupervised account matching methods (UAM): account matching according to usernames, locations, etc. We applied the unsupervised matching methods proposed in [19], but with adjusted attributes according to our datasets.
(2) Supervised account matching methods (SAM): account matching according to interest vectors. We learnt from the work [22], predicted the pair of accounts with corpus-based and knowledge-based interest vectors, respectively.
(3) Account matching based on Stable Matching (AMbSM): our proposed approach. We took both profile attributes and interest vector into consideration, and complied with the one-to-one constraint.

Evaluation measurement to evaluate the performance of account matching, we used accuracy (Acc) as follows:

$$\mathrm{Acc} = \frac{\mathrm{len}(U_{\mathrm{Correct}})}{\mathrm{len}(U_{\mathrm{Total}})}$$

where U_{Correct} represents the correct matches, U_{Total} represents all the matches, and $\mathrm{len}(U)$ is the size of U.

5.2.2 Experiment results

We separated the accounts into two sets, used one group as the training set, while the other group as the testing set. There were three kinds of data: username, gender/location, and interest vector, each of which was assigned a proper weight factor α, β, γ in [0, 1], and $\alpha + \beta + \gamma = 1$.

Firstly, we compared the five different methods, the results of which are shown in Table 1. We can see that unsupervised methods have quite good outcomes when the number of tested data is smaller, and the accuracy result degrades when the dataset

Table 1 Performance comparison of different methods for account matching

	Size = 50	Size = 100	Size = 200	Size = 300	Size = 500
UAM	0.46	0.37	0.38	0.37	0.35
SAM (corpus-based interest vector)	0.50	0.38	0.41	0.38	0.36
SAM (knowledge-based interest vector)	0.55	0.38	0.42	0.39	0.38
AMbSM (corpus-based interest vector)	0.70	0.43	0.44	0.44	0.41
AMbSM (knowledge-based interest vector)	0.65	0.45	0.46	0.46	0.42

Table 2 Performance comparison of different factors for account matching

	Size = 50	Size = 100	Size = 200	Size = 300	Size = 500
Profile attributes	0.55	0.40	0.42	0.41	0.36
Corpus-based interest vector	0.05	0.00	0.02	0.03	0.04
Knowledge-based interest vector	0.00	0.01	0.02	0.04	0.05
Profile attributes + corpus-based interest vector	0.70	0.43	0.44	0.44	0.41
Profile attributes + knowledge-based interest vector	0.65	0.45	0.46	0.46	0.42

gets larger, which is because larger username sets cause collision, as therefore reduces the differences among accounts. With the involving of historical behaviors, supervised methods could have much more better results than unsupervised ones, especially when with large datasets. AMbSM's results are better than the unsupervised and supervised methods, which supports the assumption of this paper: historical user behaviors as well as the one-to-one matching restriction will better the account matching procedure.

Secondly, we further analyzed our methods by conducting a group of experiments with different scale of datasets. The factors included profile attribute (PA), user interest in corpus, user interest in knowledge base, and Table 2 shows the experiment results. We can see that corpus-based and knowledge-based methods have better results as the scale of dataset increases, because larger datasets mean better results of the training model.

5.3 Experiments on predicting methods

5.3.1 Methodology and comparison

In this section, we tested our cross network predicting methods. We compared our methods FHSOP with a personalized recommendation method based on click behavior (HIST), a collaborative filtering method based on matrix factorization (MF) [29], both of which are popular methods for behavior prediction. Besides, to compare the accuracy of the predicting methods with the cold-start situation, we deliberately removed some accounts' purchase history, and used the popularity (POPU) [30] of the products for purchase prediction.

5.3.2 Evaluation measurement

To evaluate the event recommendation results, we adopted two standard evaluation metrics: P@K (Precision at Position *k*) and NDCG. P@K is mainly used in ranking problem. Normalized discounted cumulative gain (NDCG) measures the performance of a recommendation system based on the graded relevance of the recommended entities. It varies from 0.0 to 1.0, with 1.0 representing the ideal ranking of the entities.

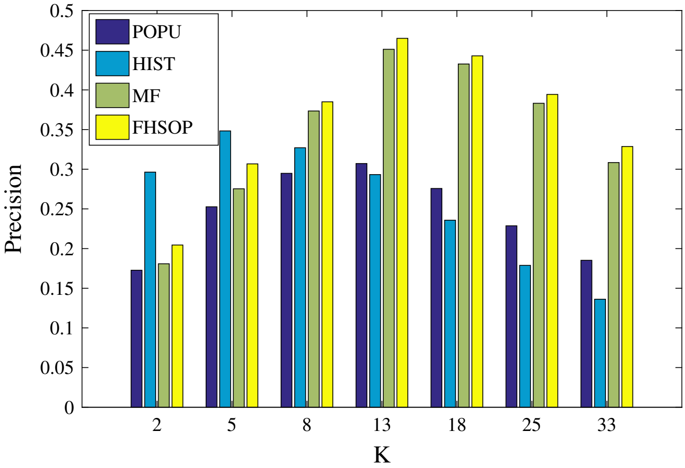

Fig. 3 Experiment results of P@K

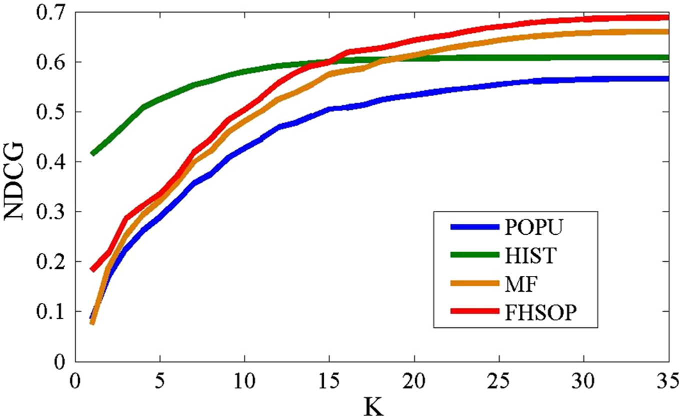

Fig. 4 Experiment results of NDCG

5.3.3 Experiment results

Figures 3 and 4 depict the predicting performances of FHSOP in comparison with POPU, HIST, and MF. From the experiment results, we can see that:

- As an online shopping site, the number of orders varies dramatically among eBay's product categories. For example, the clothing takes up to 13.08 % of the total purchase behaviors in our dataset. This confirms the necessity of involving online shopping characteristics into our predicting method.
- When K is less than a threshold (around 5–8 in this case), predicting methods based on purchase history are more accurate than other methods. However as K increases, the predicting result of FHSOP and MF significantly outperforms HIST. We can say that HIST does not contain preference information other than what

the user has already shown in his or her buying history, therefore is not able to recommend "new" things to the user.

- FHSOP considers the purchase history dimension P_H in online shopping network, and the user profile dimension P_D and user statuses dimension P_M in social network. The result shows that our method's predicting precision is around 5 % better than MF.

6 Conclusions

In this paper, we proposed a game-theoretic method to identify account pairs between social network sites and online shopping sites. We then presented a series of strategies to calculate account similarity based on account demographic difference and interest vector by applying the game version of stable matching, and extending it to solve account matching in heterogeneous networks. The experiment results show that our method identifies up to 70 % of correlation accounts between Facebook and eBay.

Moreover, we proposed FHSOP, a predicting method that combines heterogeneous social network information and online shopping information to predict the purchase behaviors. The experiment results show that by applying three predicting dimensions from both online shopping network and social network, FHSOP outperforms the state-of-the-art methods MF by 5 %.

Acknowledgments This work is supported by National Grand Fundamental Research 973 Program of China (No. 2014CB340401, 2013CB329602), National Natural Science Foundation of China (No. 61173008, 61232010, 61303244, 61402022, 61402442), and Beijing Nova Program (No. Z12110100251-2063).

References

1. Doreian P, Stokman F (2013) Evolution of social networks. Routledge, London
2. Kong X, Zhang J, Yu PS (2013) Inferring anchor links across multiple heterogeneous social networks. In: Proceedings of the 22nd ACM international conference on conference on information and knowledge management. ACM, New York
3. Chen CC, Wan YH, Chung MC, Sun YC (2013) An effective recommendation method for cold start new users using trust and distrust networks. Inf Sci 224:19–36 (Elsevier)
4. Burke R (2007) Hybrid web recommender systems. In: The adaptive web. Springer, Berlin, pp 377–408
5. Carmagnola F, Cena F (2009) User identification for cross-system personalisation. Inf Sci 179(1):16–32
6. Noam N et al (2007) Algorithmic game theory. Cambridge University Press, Cambridge
7. Barthwal R, Misra S, Obaidat MS (2013) Finding overlapping communities in a complex network of social linkages and internet of things. J Supercomput 66(3):1749–1772
8. Kim J, Kang S, Lim Y, Kim HM (2013) Recommendation algorithm of the app store by using semantic relations between apps. J Supercomput 65(1):16–26
9. Schafer JB, Frankowski D, Herlocker J, Sen S (2007) Collaborative filtering recommender systems. In: The adaptive web. Springer, Berlin, pp 291–324
10. Deshpande M, Karypis G (2004) Item-based top-*N* recommendation algorithms. ACM Trans Inf Syst 22(1):143–177

11. Jin R, Chai JY, Si L (2004) An automatic weighting scheme for collaborative filtering. In: Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval. ACM, New York, pp 337–344
12. Jannach D (2006) Finding preferred query relaxations in content-based recommenders. In: Proceedings of the 3rd international IEEE conference on intelligent systems, pp 355–360
13. Lin J, Sugiyama K, Kan MY, Chua TS (2013) Addressing cold-start in app recommendations: latent user models constructed from twitter followers. In: Proceedings of the international ACM SIGIR conference on research and development in information retrieval. ACM, New York, pp 283–292
14. Jia Y, Wang Y, Jin X, Cheng X (2014) TSBM: the temporal-spatial Bayesian model for location prediction in social networks. In: Proceedings of the international conference on web intelligence
15. Zhang YZ, Pennacchiotti M (2013) Predicting purchase behaviors from social media. In: Proceedings of the 22nd international world wide web conference. ACM, New York, pp 1521–1531
16. Buccafurri F et al (2012) Discovering links among social networks. In: Machine learning and knowledge discovery in databases. Springer, Berlin, pp 467–482
17. Bartunov S et al (2012) Joint link-attribute user identity resolution in online social networks. In: Proceedings of the workshop on social network mining and analysis, series SNA-KDD
18. Peled O et al. (2013) Entity matching in online social networks. In: Proceedings of the IEEE International Conference on Social Computing, ser. SocialCom
19. Xu J et al (2007) Complex problem solving: identity matching based on social contextual information. Faculty Publications and Research, College of Information Science and Technology, Drexel University
20. Shehab M, Ko MN, Touati H (2012) Social networks profile mapping using games. In: Proceedings of the 3rd USENIX conference on web application development, series WebApps
21. Raad E, Chbeir R, Dipanda A (2010) User profile matching in social networks. In: Proceedings of the 13th IEEE international conference on network-based information systems, series NBiS
22. Iofciu T et al (2011) Identifying users across social tagging systems. In Proceedings of the 5th international AAAI conference on weblogs and social media, series ICWSM
23. Davoodi E, Kianmehr K, Afsharchi M (2013) A semantic social network-based expert recommender system. Appl Intell 39(1):1–13
24. Liu Q, Wang Y, Li J (2014) Predicting user likes in online media based on conceptualized social network profiles. In: Proceedings of the 16th Asia–Pacific web conference
25. Aciar S, Zhang D, Simoff S (2006) Recommender system based on consumer product reviews. In: Proceedings of the 2006 IEEE/WIC/ACM international conference on web intelligence. IEEE Computer Society, New York, pp 719–723
26. Shepstone SE, Tan ZH (2013) Demographic recommendation by means of group profile using speaker age and gender recognition. In: Proceedings of the 14th annual conference of the international speech communication association
27. Jamali M (2013) HeteroMF: recommendation in heterogeneous information networks using context dependent factor models. In: Proceedings of the 22nd international conference on world wide web
28. Diao Q, Qiu M, Wu CY, Smola AJ, Jiang J, Wang C (2014) Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS). In: Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, New York
29. Salakhutdinov R, Mnih A (2007) Probabilistic matrix factorization. In: Proceedings of the 21st annual conference on neural information processing systems, NIPS07
30. Steck H (2011) Item popularity and recommendation accuracy. In: Proceedings of the 5th ACM conference on recommender systems. ACM, New York, pp 125–132