# BASS: A Bootstrapping Approach for Aligning Heterogenous Social Networks

Xuezhi Cao(✉) and Yong Yu

Apex Data and Knowledge Management Lab,
Shanghai Jiao Tong University, Shanghai, China
{cxz,yyu}@apex.sjtu.edu.cn

**Abstract.** Most people now participate in more than one online social network (OSN). However, the alignment indicating which accounts belong to same natural person is not revealed. Aligning these isolated networks can provide united environment for users and help to improve online personalization services. In this paper, we propose a bootstrapping approach BASS to recover the alignment. It is an unsupervised general-purposed approach with minimum limitation on target networks and users, and is scalable for real OSNs. Specifically, we jointly model user consistencies of usernames, social ties, and user generated contents, and then employ EM algorithm for the parameter learning. For analysis and evaluation, We collect and publish large-scale data sets covering various types of OSNs and multi-lingual scenarios. We conduct extensive experiments to demonstrate the performance of BASS, concluding that our approach significantly outperform state-of-the-art approaches.

**Keywords:** Network alignment · Heterogenous networks · User modeling

## 1 Introduction

Online social network (OSN) is playing an important role in multiple aspects of our lives. We have different OSNs for various needs, e.g. Facebook for friendship, LinkedIn for professional relations, Instagram and Pinterest for content discovery. To fully keep in touch with friends or to explore various kinds of contents, most people participate in multiple OSNs. However, the alignment indicating which accounts belong to the same natural person remains unrevealed.

Benefits of aligning OSNs include but not limited to the followings. (a) Providing an united environment for users to easily keep up-to-date with friends' online activities [22]. (b) Achieving better user modeling by aggregating action histories [25]. (c) Alleviating cold-start problem in recommender system by borrowing data from aligned networks [1,16]. (d) Providing prerequisite information for cross-network behavior analysis [12].

There are platforms trying to recover the alignment by having users manually associate their accounts, e.g. About.Me[1]. However, not all users understand the

[1] http://about.me.

P. Frasconi et al. (Eds.): ECML PKDD 2016, Part I, LNAI 9851, pp. 459–475, 2016.
DOI: 10.1007/978-3-319-46128-1_29

**Table 1.** Summary of existing approaches

| Property | Name-Based [14,23] | Profile-Based [4,16,19,22] | Network-Based [10,21] | Specific Sites [9,10] | Our Solution BASS |
|---|---|---|---|---|---|
| Target Site | general | general | social relation | tag/geo-based | general |
| Target User | similar name | with profile | general | general | general |
| Full Mapping | no | no/yes | yes | yes | no |
| Leverage UGC | no | no, but possible | no | homogeneous | heterogeneous |
| Learning Method | statistics | rule/supervised | supervised | rule/supervised | unsupervised |
| Scalability | excellent, $O(N)$ | worst at $O(N^2K)$ | poor, $O(N^2K)$ | poor, $O(N^2)$ | good, $O(ND^2)$ |

benefits and do it willingly. It is preferred if we recover the alignment automatically by mining accessible information. Attempts are made by employing information such as username [14], user profile (email, location, education) [16], [19] and social tie [21]. Due to the task's recency, limitations still exist (summarized in Table 1):

**Generality:** Limitations on target users implicitly exist in username-based and profile-based approaches. They target only at users with same/similar usernames and users with complete profile respectively. There are also works target at specific types of networks, e.g. tagging system [9] and location-based networks [10]. Besides, most works assume all users participate in both networks (full mapping assumption), i.e. the set of common users is known as prior knowledge. However, mining this information itself is not a trival task.

**Learning Method:** Beside rule-based methods, supervised learning is widely used in existing works. However, acquiring enough training data for real OSNs (10 %-30 % according to existing experiments) is impractical.

**Scalability**[2]**:** For real OSN applications, scalability must be achieved. However, only few existing works discuss this issue. By detailed analysis, several works have theoretical time complexity of over $O(N^2)$ thus not scalable for OSN scale.

In this paper, we propose BASS, a bootstrapping approach that is freed from aforementioned limitations. It captures user consistencies of usernames, social ties and preferences jointly. To model the consistencies, partial alignment is required as pre-knowledge. Instead of using training data as in traditional approaches, we model the alignment as unobserved latent variables and employ EM-fashioned algorithm for the learning, leading to an unsupervised approach. For scalability, we achieve time complexity of $O(ND^2)$, which can be considered as linear to the size of the network. Detailed comparisons are listed in Table 1.

Note that aligning social networks will not result in privacy leak. The alignment is recovered using only public available information user revealed in OSNs. In other word, such alignment already exists online just not explicitly revealed yet. For users who don't want to be aligned, understanding how their identities got revealed can be the guidance for future actions to prevent it.

The paper is organized as follows. In Sect. 2, we discuss the related works. Then we define the task in Sect. 3. We introduce the data sets and preliminary

[2] Notations: $N$-number of nodes, $D$-network degree, $K$-feature space.

analysis in Sect. 4. BASS is proposed and discussed in Sect. 5. We present the experiment results in Sect. 6. Finally we draw conclusions in Sect. 7.

## 2 Related Work

### 2.1 Social Network Alignment

Due to the flexibility of the task and the variety of information available, researchers tackle this task from different angles:

**Username.** As the identification in OSNs, it is highly valuable for this task. Zafarani et al. make several assumptions upon usernames [23]. However, they are later claimed to be false in 75.47 % cases by analysis in [14]. Liu et al. further divide the task into alias-disambiguation (differentiating accounts with same username) and alias-conflation (linking accounts with different usernames) [14]. They model alias-disambiguation as binary classification task and leave alias-conflation unsolved. However, the coverage of alias-disambiguation is limited. By our analyze only 21.52 % users have same username across networks.

**User Profile.** Vosecky et al. represent profiles as features and propose feature selection and similarity calculation accordingly [22]. Nunes et al. tackle it with classification models (SVM and Random Forest) [19]. How to handle missing data is discussed in [16]. These approaches depend highly on the information availability. However, the availability may be limited due to privacy setting or incomplete profile. As reported in [13], there is a growing trend of users' awareness of privacy. The accessibility might be further restricted. Besides, user profiles may be heterogeneous, partly missing or with false information [13], making the profile modeling harder and require heavy manually work [14].

**Social Relationship.** Friend relations and group relations are also considered [10]. Tan et al. use latent vector to capture the graph information, and then combine it with username using rule-based method [21]. The benefit of social relationship is its accessibility. As reported in [11], friends lists can be easily accessed in ten of the twelve analyzed OSNs.

There are also works focus on certain types of OSNs. Iofciu et al. aim at aligning across tagging systems [9]. Geo-location and writing style are considered in [7]. Liu. et al. take advantage of user behavior and topic modeling [15]. Despite the performance, they do not directly lead to general solutions.

Most existing works employ supervised learning technics [19,21,22]. Large amount of training data is required, mostly proportional to the network size. We need heavy manual work to apply such approaches for real OSNs.

### 2.2 Author Identification

Although the task is to some extent similar with author identification, there are still differences. Because authors mostly use real name or same pseudonym in all articles, author identification focuses more on author-disambiguation. On the other hand, in this task we also need to align accounts with different usernames. For techniques, author identification focuses more on linguistic and writing style analysis [8,26], while we need to leveraging various heterogeneous user generated contents. Further, missing information and untruthful information do not emerge in author identification for most cases. Therefore, author identification approaches can not be directly borrowed for aligning OSNs.

### 2.3 Security and Privacy

This task is also considered as a security and privacy issue [2,6,13,18]. They focus on answering whether the current OSNs are safe in the sense of anonymity protection. Thus they aim at re-identifying only a part of the users and focus on precision instead of recall. However, our goal is to recover the whole alignment. The focus also shifts toward recall and large scale.

## 3 Problem Definition

We first formulate online social networks and then define the alignment task.

**Definition 1.** *An online social network is: $\mathcal{S} = (U, E, O, P)$, where $U$ is the set of user accounts; $E$ is the set of social relations; $O(u)$ is the ownership oracle indicting who owns the account; $P(u)$ is the profile and user generated contents.*

**Definition 2.** *Social network alignment task is: Given two social network $\mathcal{S}_A, \mathcal{S}_B$, generate alignment $\hat{R} \subset U_A \times U_B$ where $(u, v) \in \hat{R}$ indicates that accounts $u, v$ belong to same natural person according to the algorithm. The ground truth is:*

$$R = \{(u, v) | u \in U_A, v \in U_B, O_A(u) = O_B(v)\} \tag{1}$$

Following this definition, we remove two constraints that widely exist in previous works. The first is one-to-one constraint [10], forcing each account to align with at most one account in the other network. The other is full mapping assumption, assuming all users participate in both networks (or the set of common users is already known).

**Preferred Properties.** Recall the existing limitations we summarized in introduction (Table 1). We prefer the solution to have the following three properties. **Generality:** Minimize assumptions on target sites and target users. **Unsupervised:** Minimize human effort needed for training. **Scalability:** Scalable to real social network scale (billion-scale).

## 4 Data Set

We collect and publish two data sets for comprehensive evaluation[3]. The data sets cover both English and Chinese sites, and include general OSNs, microblogging and movie rating sites.

[3] http://dataset.apexlab.org/bass/.

**Facebook-Twitter:** About.Me is a third party platform for associating one's accounts from different OSNs including Facebook and Twitter. We collect 1,107,695 About.Me accounts as well as the corresponding social links. For this data set we have 328,224 aligned pairs.

**Weibo-Douban:** Weibo and Douban[4] are one of China's largest microblogging and movie rating sites respectively. Alignment between them is revealed explicitly in Douban's user profile (self descriptions). In total we have 141,614 aligned users. Besides the network, we also collect movie rating histories (123.49 per user) and microblogs (343.78 per user) as user generated contents (UGCs).

### 4.1 Consistency Analysis

For further insight, we conduct analysis on user consistencies across networks.

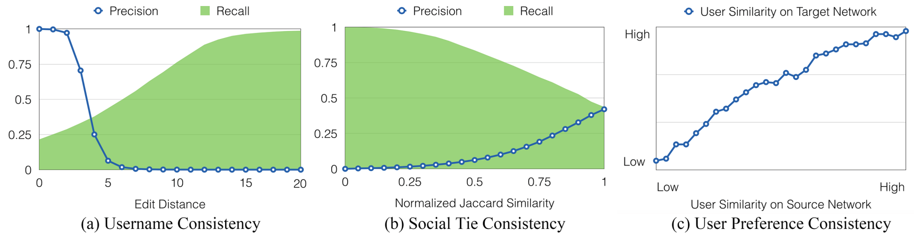
**Username Consistency.** We employ edit distance to measure similarity between usernames. Define $P_{ed}(d)$ to be the pairs of accounts with edit distance less than or equal to $d$, precision and recall as follow:

$$Prec@d = \frac{|R \cap P_{ed}(d)|}{|P_{ed}(d)|}, Rec@d = \frac{|R \cap P_{ed}(d)|}{|R|} \tag{2}$$

where $R$ is the ground truth alignment (Eq. 1). Figure 1(a) depicts the result, indicating strong relation between username and network alignment. However, the recall is limited. Only 30 % can be achieved for an acceptable precision.

**Social Tie Consistency.** Social links in OSNs reflect user's social ties or interests to some extent. We demonstrate social tie consistency by analyzing whether one's social relations in different OSNs tend to be overlapping. We use Jaccard Similarity Coefficient to capture the overlapping level. As only the relative value is required, we normalize the coefficient according to each user.

$$J(u,v) = \frac{O_A(E_A(u)) \cap O_B(E_B(v))}{O_A(E_A(u)) \cup O_B(E_B(v))}, J_n(u,v) = \frac{J(u,v)}{\max_{v' \in U_B} J(u,v')} \tag{3}$$



(a) Username Consistency
(b) Social Tie Consistency
(c) User Preference Consistency

**Fig. 1.** Consistency across OSNs (Username, Social Tie and User Preference)

[4] http://www.weibo.com/, http://www.douban.com/.

where $E_A(u), E_B(u)$ are the neighbors of $u$ in network $A, B$ respectively and $O_A, O_B$ are the ownership oracles. Result in Fig. 1(b) indicates the existence of social tie consistency. However, it also indicates that even the alignment is given except the target pair, precision based only on social tie is not ideal ($\sim 40\,\%$).

**User Preference Consistency.** We demonstrate user preference consistency by showing that users with similar UGCs in one network tend to be similar in the other. Specifically, we employ topic model (LDA [3]) for modeling text-based UGCs and Jaccard Similarity Coefficient for item-based UGCs (rating/purchasing logs). Result in Fig. 1(c) supports the assumption that user preference consistency exists.

## 5 Bootstrapping Approach

In this section we propose our approach BASS. We first present the work flow, then discuss the algorithm details, and finally tackle the scalability issues.

### 5.1 Work Flow

We aim at recovering the alignment by mining consistencies of usernames, social ties and user preferences across the networks. However, we need partial alignment as pre-knowledge to model such consistencies. Specifically, to model social ties we need the alignment over target user's friends, and to model user preferences we need large scale aligned pairs to learn the preference transfer between heterogeneous networks. Traditionally, researchers employ labeled training data to solve this, leading to supervised approach. Instead, we model the alignment as unobserved latent data along with the consistency model, and then employ Expectation-Maximization algorithm for the parameter learning. Following this, we achieve an unsupervised bootstrapping approach.

We have two sub-models in our framework. One is the consistency model $\mathcal{C}(X, R)$ that captures the aforementioned consistencies based on observed data $X$ as well as the given alignment $R$, where $X$ contains usernames, social relations and user generated contents in both networks. The other is the classification model $\mathcal{Y}$ that takes in the features generated by $\mathcal{C}(X, R)$ and estimates the probability that two accounts belong to the same natural person. For notations, we use $\Theta = \{\Theta_\mathcal{C}, \Theta_\mathcal{Y}\}$ for the set of unknown parameters for the two parts. Our goal is to estimate the parameters $\Theta$ by maximizing the likelihood $L(\Theta; X)$, and then recover the alignment $\hat{R}$ based on the estimated parameters $\hat{\Theta}$.

$$\begin{aligned} L(\Theta; X) &= p(X|\Theta) = \sum_R p(X, R|\Theta) \\ p(X, R|\Theta) &= \prod_{(u,v)\in R} \mathcal{Y}(\mathcal{C}_{u,v}(X, R, \Theta_\mathcal{C}), \Theta_\mathcal{Y}) \prod_{(u,v)\notin R} (1 - \mathcal{Y}(\mathcal{C}_{u,v}(X, R, \Theta_\mathcal{C}), \Theta_\mathcal{Y})) \\ \hat{\Theta} &= \arg\max_\Theta L(\Theta; X), \;\; \hat{R} = \arg\max_R p(X, R|\hat{\Theta}) \end{aligned} \tag{4}$$

By viewing the alignment $R$ as the unobserved latent data, we can employ Expectation-Maximization algorithm for the learning process. For E-step, we update the alignment $R$ based on current parameters. And for M-step, we update the parameters for both consistency model and classification model based on the just-computed $R$. The overall work flow is depicted in Fig. 2.

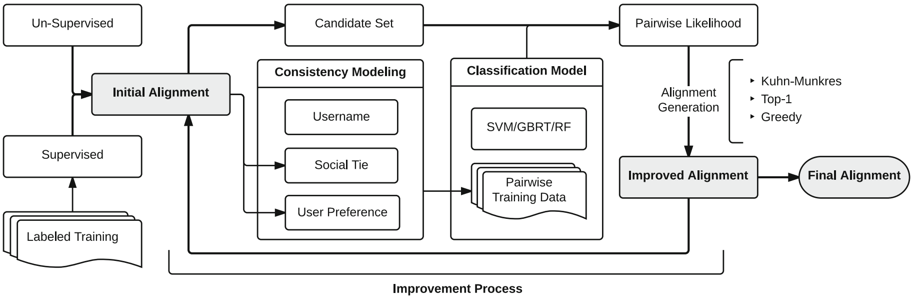


**Fig. 2.** Work Flow of **BASS**: **B**ootstrapping approach for **A**ligning **S**ocial network**S**

The benefit of bootstrapping strategy is multi-folded. Firstly, we achieve unsupervised approach thus large-scale labeled training data is no longer required. Secondly, we employ the whole network for training instead of only the labeled pairs in traditional approaches. Thirdly, we can directly adopt the bootstrapping approach for incremental online learning by considering new users as unaligned, therefore is suitable for real OSN application.

### 5.2 Algorithm Details

The key components are consistency model and classification model for M-step, and alignment generation for E-step. We first discuss the components and then extend the approach for unsupervised learning in the following subsections.

**Consistency Model.** Based on previous analysis, we target at consistencies of username, social tie and user generated content.

Username is the easiest as it doesn't depend on the alignment. Consistency features include exact/substring match, edit distance and naming patterns [24].

Social tie comes next. It depends on the alignment but fortunately is of homogeneous format (consider all relationships as undirected). We extend Jaccard Similarity Coefficient to capture the social tie consistency. Given current alignment $\hat{R}$, it can be defined by:

$$\begin{aligned} com(u,v,\hat{R}) &= (E_A(u) \times E_B(v)) \cap \hat{R} \\ J(u,v,\hat{R}) &= \frac{|com(u,v,\hat{R})|}{|E_A(u)| + |E_B(v)| - |com(u,v,\hat{R})|} \end{aligned} \tag{5}$$

The most challenging one is modeling the user generated contents consistency (or user preferences consistency) because it depends on current alignment and is heterogeneous across networks. Most UGCs can be categorized by text-based and item-based ones. Text-based UGCs include microblogging, forum and etc., and item-based ones include rating log, purchase log and etc. Therefore we target at these two types of UGCs. We model them using multi-modal Latent Dirichlet Allocation [3], where each modal corresponds to the UGCs in one network. The model is depicted in Fig. 3. The detailed distributions for upper modal (text-based site Weibo) of the multi-modal model are as below:

$$\theta_i \sim Dir(\alpha), \quad z_{ij}^w \sim Multi(\theta_i), \ \phi_k^w \sim Dir(\beta^w), \quad w_{ij} \sim Multi(\phi_{z_{ij}}^w) \tag{6}$$

And the inference is as follows:

$$P(z_{ij}^w = k | z_{\neg ij}^w, w, \phi^w, \cdot) \propto \frac{n_{ik}^{w,\neg ij} + \alpha_k}{\sum_q (n_{iq}^{w,\neg ij} + \alpha_q)} \cdot \frac{m_{kw_{ij}}^{w,\neg ij} + \beta_{w_{ij}}}{\sum_{w'} (m_{kw'}^{w,\neg ij} + \beta_{w'})} \tag{7}$$

where $n_{ik}^w$ is the number of times topic $k$ being assigned to user $i$ (number of times $z_{i*}^w = k$) and $m_{kj}^w$ is the number of times word $j$ being assigned to topic $k$. The distribution as well as the inference for lower modal with $z^m, m, \phi^m$ are similar with above. After sufficient sampling iterations, the preference distribution $\theta_i$ can be estimated by:

$$\hat{\theta}_{ij} = \frac{n_{ik}^w + n_{ik}^m + \alpha_k}{\sum_q (n_{iq}^w + n_{iq}^m + \alpha_q)} \tag{8}$$

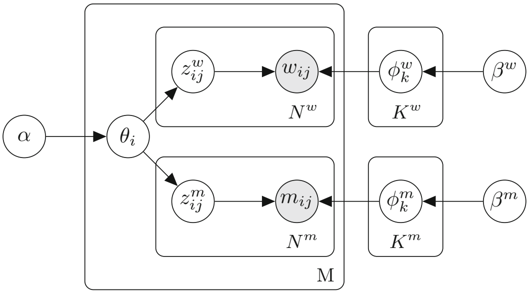


**Fig. 3.** Multi-modal topic model

The learning process of the multi-modal topic model is as follows. We consider each alignment $(u, v)$ in current alignment $\hat{R}$ as a user instance with actions in both modals. With these as anchor links, we learn the correlation and transfer between heterogeneous modals. We also consider each account $u \in U_A$ and $v \in U_B$ as a user instance with actions only in one modal.

With the multi-modal topic model, we can embed each account in both networks into the universal topic space ($\theta$). Then we can quantify the consistency between accounts $u, v$ using cosine similarity, L1 distance and Kullback-Leibler Divergence over $\theta_u$ and $\theta_v$.

**Classification Model.** We use binary classification model to separate pairs of accounts by whether they belong to same natural person. Specifically, we employ Support Vector Machine [20] here in BASS.

Following the likelihood function in Eq. (4), the current alignment $\hat{R}$ serves as the ground truth when updating the classification model. Specifically, all pairs of accounts that are aligned in the current alignment ($(u, v) \in \hat{R}$) are considered as positive instances and the rest as negative instances. Following this, we have $O(N^2)$ training instances, where $N$ is the number of accounts. Such amount is too large for scalability. We will return to this issue in Sect. 5.3.

Note that the current alignment $\hat{R}$ is not the actual ground truth. Because the classification model is considered to be noise-robust, applying it to the noisy alignment $\hat{R}$ can still optimizing the likelihood function. As long as the likelihood is being optimized, the bootstrapping approach can work properly.

**Alignment Generation.** With the consistency model and classification model trained, we can estimate the pairwise likelihood by $S(u, v) = \mathcal{Y}(\mathcal{C}_{u,v}(X, R, \Theta_{\mathcal{C}}), \Theta_{\mathcal{Y}})$. Therefore, the remaining task is: Given pairwise score $S$ where $S(u, v)$ indicates the likelihood of accounts $u, v$ belonging to the same natural person, generate the required alignment $R$.

If we align each account to at most one corresponding account, the alignment is actually a partial one-to-one mapping $M : E_A \to \{E_B \cup \perp\}$, where $M(u) = \perp$ indicates no corresponding account for $u$. The objective function is:

$$\begin{aligned} M^* &= \arg\max_M \prod_u S(u, M(u)) \\ R^* &= \{(u, M^*(u)) | u \in E_A, M^*(u) \neq \perp\} \end{aligned} \tag{9}$$

where we define $S(u, \perp) = \tau$ as the penalty for mismatch.

By considering each account as a node and $\log S(u, v) - \log \tau$ to be the weight of the edge between node $u$ and $v$, this task can be deduced to a maximum matching problem on weighted bipartite graph. Such problem can be solved perfectly in $O(N^4)$ by using Kuhn-Munkres (KM) algorithm [17], also known as Munkres assignment algorithm. The algorithm is later improved by Karp et al., achieving a time complexity of $O(N^3)$. However, this is still not scalable for real social network scale. Compromise must be made by using alternative algorithms instead of perfect matching. We discuss it later in Sect. 5.3.

**Extend to Unsupervised Version.** To minimize human effort needed, it is preferred that the algorithm can run in an unsupervised manner. In other words, the initial alignment need to be automatically generated instead of using labeled training data. In BASS, the initial alignment serves as seeds for alignment propagation. Therefore, precision is strongly required while recall is not. Previous analysis show that alignment generated by username matching fulfill the requirement and can be considered as the initial alignment.

There is another potential issue. As BASS uses current alignment to generate training data for the classification model, there might be a chance that the classification model converge to the rules that we used to generate the initial alignment. To prevent this from happening, we introduce noises into the data intentionally during training process. Specifically, we randomly alter some of the usernames (10 % in experiment) during the consistency modeling process for initial alignment so that username features do not fully suppress other features. Experiments show that the performance is not very sensitive to this parameter if set within reasonable ranges.

### 5.3 Scalability Issue

Due to tremendous size of OSN users, scalability must be considered. It is normal for social network to be of billion-scale, so even $O(N^2)$ approach is impractical. As stated previously, the size of pairwise training samples and the time complexity of matching algorithm are not scalable under current model settings. Besides, there is a hidden violation that we have $O(N^2)$ candidate pairs.

**Training Data Subsampling.** Based on learning process proposed previously, a training set of size $O(N^2)$ will be generated, with $O(N)$ positive samples and $O(N^2)$ negative ones. Therefore, we subsample over the negative samples and keep only a constant number ($k$) of negative samples for each account.

Conducting the subsampling effectively is not trival. Because most negative samples can be easily separated from the positive ones and provide almost no valuable information, purely random sample would highly jeopardized the performance. We follow the idea that boundary samples, the ones that cannot be easily separated by the classification model, are more valuable for the training process. Similar ideas are also used in other scenarios. For example, in Support Vector Machine [20], support vectors are actually boundary samples.

By assuming the models in two consecutive iterations are similar, we consider the negative samples with high but not top scores in previous iteration as boundary samples. The final approach is: for each account $u$, consider its current alignment $(u, v) \in \hat{R}$ as positive sample, and $(u, v')$ as a negative sample for all $v'$ ranked in top $k$ according to $S(u, v')$ in last iteration. Where $k$ is the parameter to balance between performance and scalability. By setting $k$ to infinity, the process degenerates to original version.

**Candidate Pair Generation.** As we cannot consider all $N^2$ pairs due to scalability concern, candidate set must be generated. Analysis in Fig. 1(b) shows that for almost all true alignment ($> 99\,\%$), their friendships overlap to some extent (with Jaccard coefficient $> 0$). Therefore, we consider only accounts pairs with common neighbors according to current alignment $\hat{R}$ as candidates. Formally we construct candidate set by $C = \bigcup_{(u,v)\in\hat{R}} E_A(u) \times E_B(v)$. Following this we obtain $O(ND^2)$ candidate pairs, where $D$ is the degree of the network. Note that no matter how the network grows, the degrees of normal users are still limited. So $O(ND^2)$ can be considered as linear to the network's size.

**Alternative Alignment Generation.** Although perfect alignment can be achieved, it requires large amount of computation. Therefore, we need efficient alternative alignment generation method. Here we propose two candidates:

**Top-1 Alignment.** Match each user account $u$ to $v$ that maximize $S(u, v)$:

$$\hat{R} = \{(u, \arg\max_{v} S(u, v))\} \cup \{(\arg\max_{u} S(u, v), v)\} \tag{10}$$

Similar as previous, we ignore alignment with $S(u, v) < \tau$. For users with no alignment, we consider them as single-site users.

**Stable-Marriage Alignment.** A matching between two sets of elements is a stable marriage matching if there does not exist a pair of elements that both elements prefer each other than their current alignment. We borrow this definition for social network alignment problem. The algorithm for original stable marriage problem is: first selecting an unaligned element $u$ and its most preferred element $v$ that $u$ has not proposed yet; if $v$ is available then link $(u, v)$; otherwise if $v$ also prefers $u$ over its current alignment then link $(u, v)$ and release $v$'s original alignment. This algorithm runs in time complexity of $O(N^2)$. Fortunately, in this setting we have another property: the preference matrix is symmetric. This property enables us to further speed up the computation. Specifically, if we traverse the candidate pairs $(u, v)$ in descend order of $S(u, v)$ instead of randomly selecting $u$, we will never need to replace existing alignment as in the traditional algorithm. Thus we can achieve time complexity of $O(|C| \log |C|)$ where $C$ is the candidate set. Similar as previous, we link only when $S(u, v) > \tau$.

## 6 Experiments

### 6.1 Experiment Setting

The data sets are discussed in Sect. 4. Recall the data includes general-purposed, microblogging and movie review OSNs, and covers multi-lingual (English and Chinese) scenarios. Now we explain the metrics and comparing algorithms.

**Evaluation Metrics:** As defined in Sect. 3, the task is a retrieval task that mines the aligned pairs of accounts. F-1 Score is used as the metric, defined by:

$$Prec(\hat{R}) = \frac{|\hat{R} \cap R|}{|\hat{R}|},\ Recall(\hat{R}) = \frac{|\hat{R} \cap R|}{|R|},\ F_1(\hat{R}) = 2 \cdot \frac{Prec(\hat{R}) Recall(\hat{R})}{Prec(\hat{R}) + Recall(\hat{R})} \tag{11}$$

where $R$ is the ground truth alignment (Eq. (1)) and $\hat{R}$ is the prediction.

**Comparing Algorithms:** We compare with the state-of-the-art approaches that based on similar information with BASS (username, social tie and user preference), which are listed and described as follows:

**BASS:** The bootstarpping approach proposed in this paper. Support Vector Machine (LibSVM [5]) is employed as the classification model. Stable-Marriage alignment is used unless indicated otherwise.
**BASS-H:** BASS without UGC modeling (user preference consistency).
**BASS-U:** Unsupervised version of BASS (Sect. 5.2).
**MNA:** Multi-Network Anchoring proposed by Kong et al. in [10].
**MAH:** The Manifold Alignment on Hypergraph approach, by Tan et al. [21].
**MOBIUS:** Aligning by modeling user behaviors, proposed by Zafarani in [24].
**NAME:** Aligning accounts with same username. Precision for exact name matching is almost 1, the only concern is coverage. So it can be seemed as an upper-bound for works focusing on alias-disambiguation [14].

There are also approaches not compared due to limitations on target sites or target users (tagging system [9], profile-based [4,16,19,22] and etc.).

### 6.2 Results and Analysis

We first conduct experiments based on the full mapping assumption (the assumption in most existing works): all users participate in both networks thus a perfect alignment exists. 30 % of alignment is given as training data (except for the unsupervised version). Parameters such as $\alpha, \beta, \tau$ are set by cross validation.

We show the results in Table 2. Our approaches, both supervised and unsupervised versions, achieve significantly better performance comparing to state-of-the-art algorithms. Note that UGCs are only available in Weibo-Douban data set, therefore BASS-H and BASS are the same over Facebook-Twitter data set. Precision and recall of MAH and MOBIUS are the same respectively because they always produce full mapping, and in this experiment setting a perfect mapping exists (full mapping assumption).

Note that as OSN varies, experimental results using different data sets are not numerically comparable. For example, In our data set only 21 % users share same username while 52 % in data used in [21]. The differences may due to the collecting methodology and the data source used. As previous works did not

**Table 2.** Experimental results with 30 % training data

| Approach | Facebook-Twitter | | | Weibo-Douban | | |
|---|---|---|---|---|---|---|
| | Precison | Recall | F1-Score | Precison | Recall | F1-Score |
| BASS | 84.40 % | **79.75 %** | **0.8201** | 79.49 % | **76.22 %** | **0.7782** |
| BASS-H | 84.40 % | 79.75 % | 0.8201 | 76.56 % | 73.26 % | 0.7487 |
| BASS-U | 82.52 % | 78.10 % | 0.8025 | 77.64 % | 74.88 % | 0.7623 |
| MAH | 42.82 % | 42.82 % | 0.4282 | 41.74 % | 41.74 % | 0.4174 |
| MNA | 67.64 % | 62.21 % | 0.6481 | 64.39 % | 61.41 % | 0.6287 |
| MOBIUS | 55.48 % | 55.48 % | 0.5548 | 51.37 % | 51.37 % | 0.5137 |
| NAME | **100.00 %** | 21.52 % | 0.3541 | **100.00 %** | 14.70 % | 0.2562 |

publish their data, we can only compare the approaches using our data (we also publish the data sets to the community). The scale of the data set also contribute to the difficulty of aligning. It is rather difficult to find the corresponding accounts from millions of accounts comparing to from hundreds of accounts.

**Heterogeneous UGC Modeling:** By comparing BASS and BASS-H in Table 2, we show that UGC modeling do improve the alignment quality. To gain further insight, we list part of the resulting word-dictionary along with the movie-dictionary in Table 3. Correlation can be noticed, indicating the multimodal LDA can capture the heterogeneous UGCs and embed users by their underlying general preferences. For example, the first topic indicates the users tweet about pets are more likely to enjoy comedies, cartoons etc.

**Effect of Training Size:** We vary the size of training size from 10 % to 50 %. Results showed in Table 4 indicate that our unsupervised version BASS-U defeats the existing supervised approaches even when 50 % data is given. It is also noticeable that the first 20 % data does not result in great improvement in BASS comparing to BASS-U, indicating that using unsupervised approach can capture most common knowledge for the aligning task.

**Single Network Users:** Most existing works assume that all users participate in both networks (full mapping assumption). However, it is not the real-world scenario. Now we break this limitation and expand the data set by adding users that participate in only one OSN. To make it more challenging, we add friends of existing users instead of purely random users. We show the results in Table 4 (right part), where $p$ indicates the ratio of users participate in both networks (traditional approaches still aim at aligning all users). Decreasing $p$ makes the task more challenging. Note that it has more impact on approaches based on social ties or user preferences, while less on ones based on only usernames (performance drops dramatically on MAH and MNA while slightly on MOBIUS). Because of the comprehensiveness of our consistency model (with no heavy dependency on specific aspect), the performance remains mostly the same with only minor drop.

**Improvement Process:** The effect of the iterative bootstrapping is depicted in Fig. 4(a). A clear trend of improvement, or the snowball effect, can be noticed in both supervised and unsupervised approaches. Note that only few rounds are needed before convergence, so it does not raise a complexity issue.

**Social Tie Strength:** As the social tie varies from site to site, we are interested in how social tie strength effects the performance of aligning. For comprehensive analysis, we generate synthetic data based on real data. We first union the social relations from both networks and consider them as the potential links. Then we plug each link into the two networks with probability of $p, q$ respectively. Here $p, q$ are parameters controlling the density and coupling strength. User profiles as well as UGCs are kept the same as the real data. By varying $p, q$ from 0 to 1, we can simulate different social tie scenarios. Specifically, we have: (1) *Isomorphic* by setting $p = q = 1$; (2) *Subnetwork* by setting $p = 1, q < 1$ or vice versa; (3) *Partially Overlapping* by setting $p, q < 1$.

**Table 3.** Dictionaries for Multi-modal topic model over Weibo & Douban

| ID | Top Words | Top Movies |
|---|---|---|
| 1 | Food, Cat, Friend, Dog, Home, Like, Life | Hotaru no haka, The Pursuit of Happyness, Jeux d'enfants The Devil Wears Prada, Up, Tonari no Totoro, Ratatouille |
| 2 | China, Article, Book, Issue, America, Country, Society | Inception, Social Network, Source Code, Avatar, WALL·E V for Vendetta, The Lord of the Rings, Argo The Shawshank Redemption, The Bourne Identity, Titanic |
| 3 | We, Love, Myself, Life, Want, Time, Like, World | Amour, Love Letter, Amlie, Forrest Gump, Before Sunrise Before Sunset, Flipped, Love Actually, The Notebook |

**Table 4.** Varying size of Training Data & Common User Ration (results in Table 4 are according to Facebook-Twitter set.)

| F1 Score | Training Size | | | | | Common User Ration $p$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 10 % | 20 % | 30 % | 40 % | 50 % | 100 % | 90 % | 80 % | 70 % | 60 % |
| MAH | 0.3789 | 0.4065 | 0.4282 | 0.4330 | 0.4426 | 0.4282 | 0.4026 | 0.3706 | 0.3444 | 0.3253 |
| MNA | 0.6021 | 0.6259 | 0.6481 | 0.6549 | 0.6612 | 0.6481 | 0.6201 | 0.5928 | 0.5503 | 0.5196 |
| MOBIUS | 0.5327 | 0.5457 | 0.5548 | 0.5617 | 0.5620 | 0.5548 | 0.5479 | 0.5465 | 0.5431 | 0.5390 |
| BASS | 0.8073 | 0.8093 | 0.8201 | 0.8265 | 0.8374 | 0.8201 | 0.8205 | 0.8164 | 0.8158 | 0.8087 |
| BASS-U | 0.8025 | | | | | 0.8025 | 0.8047 | 0.7985 | 0.7990 | 0.7973 |

Results for the $p = q$ cases are showed in Fig. 5(a). Our approaches out perform existing approaches except when $p, q$ are too small, which indicates almost no social relations available. In these cases the social ties become total noises and the consistency propagation may be restricted. Hence, username-based approaches have great advantage. We depict results for all $p, q$ in Fig. 5(b). Except for the cases that one network's social relationship is too weak ($\min(p, q) \leq 0.1$), our approach has a satisfying performance.

### 6.3 Performance vs Scalability

**Subsampling.** Recall that we conduct subsampling over training data for the classification model. A parameter $k$ controls the number of negative samples for each user. Larger $k$ leads to larger computational complexity but better performance. By varying $k$, we show the results in Fig. 4(b). We conclude that when $k$ is larger than some small threshold (3–5), keep increasing $k$ does not achieve a significantly better performance. This indicates that our subsampling

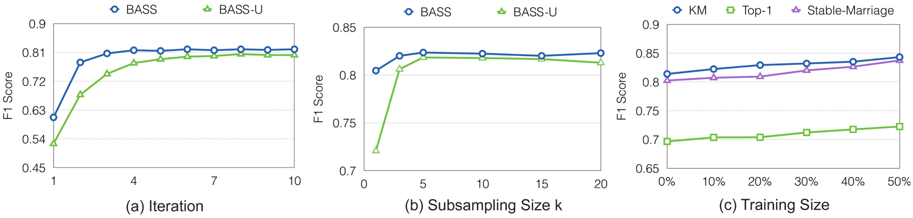


**Fig. 4.** Detail analysis - Iteration, Alignment Algorithm, Subsampling

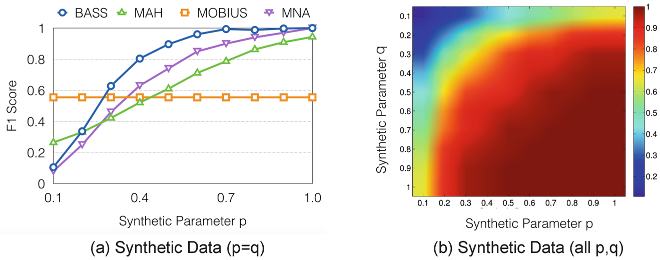


**Fig. 5.** Experimental results over synthetic data

strategy can shrink the training set of size $O(N^2)$ into of size $O(N)$ with ignorable sacrifice of performance.

**Alignment Algorithms.** Due to the high time complexity of KM algorithm, two alternative alignment algorithms (Top-1 alignment and Stable-Marriage alignment) are proposed. We show the results using different alignment algorithms in Fig. 4(c). As expected, KM algorithm always gives the best result. Stable-Marriage alignment can always achieve a compatible result with a much lower time complexity. Thus a great balance between the scalability and performance can be achieved using it.

## 7 Conclusions and Future Works

In this paper, we focus on aligning heterogeneous social networks. To tackle the problem, we propose a bootstrapping approach BASS, which starts with an unperfect alignment and refines it iteratively based on consistency propagation over usernames, social ties and user preferences. The advantage is multi-folded. Firstly, it is a general-purpose approach with minimum limitation on target sites and users, and can be adopted for various heterogeneous scenarios. Secondly, full-mapping constraint is removed. Thirdly, we achieve unsupervised approach. Finally, it is scalable for large-scale social networks without jeopardizing the performance. We also collect and publish large-scale real-world data sets covering various scenarios. To the best of our knowledge, this is the first public available data set for this task. We conduct comprehensive experiments. Results indicate that BASS outperform state-of-the-art approaches with a relative improvement of about 40 % in most scenarios.

Due to the novelty of this topic, there exist plenty of future works. One direction is considering aligning accounts over multiple social networks instead of two, such task would be much more general but harder as well. Further, we can employ the aligned social networks for user behavior analysis across sites. It is also an interesting topic to reveal the underlying real friend relation, i.e. the true friendship among natural persons. Following this work, we can also study what online actions jeopardies the user's anonymity and how to prevent accordingly.

## References

1. Abel, F., Henze, N., Herder, E., Krause, D.: Interweaving public user profiles on the web. In: De Bra, P., Kobsa, A., Chin, D. (eds.) UMAP 2010. LNCS, vol. 6075, pp. 16–27. Springer, Heidelberg (2010). doi:10.1007/978-3-642-13470-8_4
2. Backstrom, L., Dwork, C., Kleinberg, J.: Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In: WWW, pp. 181–190. ACM (2007)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
4. Carmagnola, F., Cena, F.: User identification for cross-system personalisation. Inf. Sci. **179**(1), 16–32 (2009)
5. Chang, C.C., Lin, C.J.: Libsvm: a library for support vector machines. ACM Trans. Intell. Syst. Technol. (TIST) **2**(3), 27 (2011)
6. Frankowski, D., Cosley, D., Sen, S., Terveen, L., Riedl, J.: You are what you say: privacy risks of public mentions. In: SIGIR, pp. 565–572. ACM (2006)
7. Goga, O., Lei, H., Parthasarathi, S.H.K., Friedland, G., Sommer, R., Teixeira, R.: Exploiting innocuous activity for correlating users across sites. In: WWW, pp. 447–458. International World Wide Web Conferences Steering Committee (2013)
8. Houvardas, J., Stamatatos, E.: N-gram feature selection for authorship identification. In: Euzenat, J., Domingue, J. (eds.) AIMSA 2006. LNCS (LNAI), vol. 4183, pp. 77–86. Springer, Heidelberg (2006). doi:10.1007/11861461_10
9. Iofciu, T., Fankhauser, P., Abel, F., Bischoff, K.: Identifying users across social tagging systems. In: ICWSM (2011)
10. Kong, X., Zhang, J., Yu, P.S.: Inferring anchor links across multiple heterogeneous social networks. In: CIKM. ACM (2013)
11. Krishnamurthy, B., Wills, C.E.: On the leakage of personally identifiable information via online social networks. In: Proceedings of the 2nd ACM Workshop on Online Social Networks, pp. 7–12. ACM (2009)
12. Kumar, S., Zafarani, R., Liu, H.: Understanding user migration patterns in social media. In: AAAI (2011)
13. Labitzke, S., Taranu, I., Hartenstein, H.: What your friends tell others about you: low cost linkability of social network profiles. In: Proceeding of 5th International ACM Workshop on Social Network Mining and Analysis, San Diego (2011)
14. Liu, J., Zhang, F., Song, X., Song, Y.I., Lin, C.Y., Hon, H.W.: What's in a name?: an unsupervised approach to link users across communities. In: WSDM, pp. 495–504. ACM (2013)

15. Liu, S., Wang, S., Zhu, F., Zhang, J., Krishnan, R.: Hydra: Large-scale social identity linkage via heterogeneous behavior modeling. In: SIGMOD (2014)
16. Malhotra, A., Totti, L., Meira Jr., W., Kumaraguru, P., Almeida, V.: Studying user footprints in different online social networks. In: Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012), pp. 1065–1070. IEEE Computer Society (2012)
17. Munkres, J.: Algorithms for the assignment and transportation problems. J. Soc. Ind. Appl. Math. **5**(1), 32–38 (1957)
18. Narayanan, A., Shmatikov, V.: De-anonymizing social networks. In: 30th IEEE Symposium on Security and Privacy, pp. 173–187. IEEE (2009)
19. Nunes, A., Calado, P., Martins, B.: Resolving user identities over social networks through supervised learning and rich similarity features. In: Proceedings of the 27th Annual ACM Symposium on Applied Computing, pp. 728–729. ACM (2012)
20. Suykens, J.A., Vandewalle, J.: Least squares support vector machine classifiers. Neural Process. Lett. **9**(3), 293–300 (1999)
21. Tan, S., Guan, Z., Cai, D., Qin, X., Bu, J., Chen, C.: Mapping users across networks by manifold alignment on hypergraph. In: AAAI (2014)
22. Vosecky, J., Hong, D., Shen, V.Y.: User identification across multiple social networks. In: First International Conference on Networked Digital Technologies, NDT 2009, pp. 360–365. IEEE (2009)
23. Zafarani, R., Liu, H.: Connecting corresponding identities across communities. In: ICWSM (2009)
24. Zafarani, R., Liu, H.: Connecting users across social media sites: a behavioral-modeling approach. In: SIGKDD, pp. 41–49. ACM (2013)
25. Zhang, J., Kong, X., Yu, P.S.: Transferring heterogeneous links across location-based social networks. In: WSDM, pp. 303–312. ACM (2014)
26. Zheng, R., Li, J., Chen, H., Huang, Z.: A framework for authorship identification of online messages: writing-style features and classification techniques. JASIST **57**(3), 378–393 (2006)