PCT: Partial Co-Alignment of Social Networks

Jiawei Zhang
University of Illinois at Chicago
Chicago, IL, USA
jzhan9@uic.edu

Philip S. Yu
University of Illinois at Chicago, IL, USA
Institute for Data Science
Tsinghua University, Beijing, China
psyu@cs.uic.edu

ABSTRACT

People nowadays usually participate in multiple online social networks simultaneously to enjoy more social network services. Besides the common users, social networks providing similar services can also share many other kinds of information entities, e.g., locations, videos and products. However, these shared information entities in different networks are mostly isolated without any known corresponding connections. In this paper, we aim at inferring such potential corresponding connections linking multiple kinds of shared entities across networks simultaneously. Formally, the problem is referred to as the network "Partial Co-alignmenT" (PCT) problem. PCT is an important problem and can be the prerequisite for many concrete cross-network applications, like social network fusion, mutual information exchange and transfer. Meanwhile, the PCT problem is also very challenging to address due to various reasons, like (1) the heterogeneity of social networks, (2) lack of training instances to build models, and (3) *one-to-one* constraint on the correspondence connections. To resolve these challenges, a novel unsupervised network alignment framework, UNICOAT (UNsupervIsed COncurrent AlignmenT)), is introduced in this paper. Based on the heterogeneous information, UNICOAT transforms the PCT problem into a joint optimization problem. To solve the objective function, the *one-to-one* constraint on the corresponding relationships is relaxed, and the redundant non-existing corresponding connections introduced by such a relaxation will be pruned with a novel network co-matching algorithm proposed in this paper. Extensive experiments conducted on real-world co-aligned social network datasets demonstrate the effectiveness of UNICOAT in addressing the PCT problem.

Keywords

Partial Network Co-Alignment, Multiple Heterogeneous Social Networks, Unsupervised Learning, Data Mining

1. INTRODUCTION

Looking from a global perspective, the landscape of online social networks is highly fragmented. A large number of online social networks have appeared and achieved prosperous developments in recent years. Some of these networks can even provide very comparable network services and are of similar network structures. For instance, (1) Foursquare and Yelp (two famous location-based social networks) can both offer location related services for users; (2) Amazon and Ebay are both created for online e-commerce; (3) Kickstarter[1] and Indiegogo[2] are both constructed to accumulate funding for projects from the public; and (4) Youtube and Vimeo [3] both provide large amounts of video resources for users to either watch or share with friends.

In such an age of online social media, users usually participate in multiple social networks simultaneously to enjoy more social networks services, who can act as bridges connecting different networks together. Besides these common users, social networks offering similar services can also share other common information entities, e.g., locations shared between Foursquare and Yelp, and products sold in both Amazon and Ebay. Formally, the shared information entities in different networks can act as anchors aligning these networks, which can be formally named as anchor instances (e.g., the shared users can be called anchor users, while the shared locations and products can be called anchor locations and anchor products respectively). What's more, the corresponding relationships between the anchor instances (indicating they are the same information entities) across networks can be called anchor links. For instance, the corresponding relationships between the shared users can be named as user anchor link, while those between shared locations can be called location anchor link. However, in the real-world, anchor instances in different networks are mostly isolated without any known anchor links connecting them.

Problem Studied: In this paper, we want to infer different categories of anchor links connecting various anchor instances across social networks simultaneously, which is formally defined as the network "Partial Co-alignmenT" (PCT) problem. PCT is a general research problem and can be applied to different types of social networks, like Foursquare and Yelp, Amazon and Ebay. Meanwhile, as shown in Figure 1, in this paper, we will mainly focus on the partial co-alignment of location based social networks via shared users and locations with the various connection and attribute information available in the networks. PCT is an important research problem and can be the prerequisite for many concrete real-world applications, like network fusion [33, 29, 11, 31, 19], cross-network recommendation [27, 28, 34, 19], mutual community detection [32, 30], and inter-network information diffusion [26].

Besides its importance, PCT is also a novel problem and totally different from existing works on entity matching and network alignment, like (1) "*supervised anchor link inference*" [11], which focuses on inferring the user anchor links only with a supervised learning method; (2) "*user matching across networks*" [25], which

WWW 2016, April 11–15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4143-1/16/04.
http://dx.doi.org/10.1145/2872427.2883038.

[1] https://www.kickstarter.com

[2] https://www.indiegogo.com

[3] https://vimeo.com

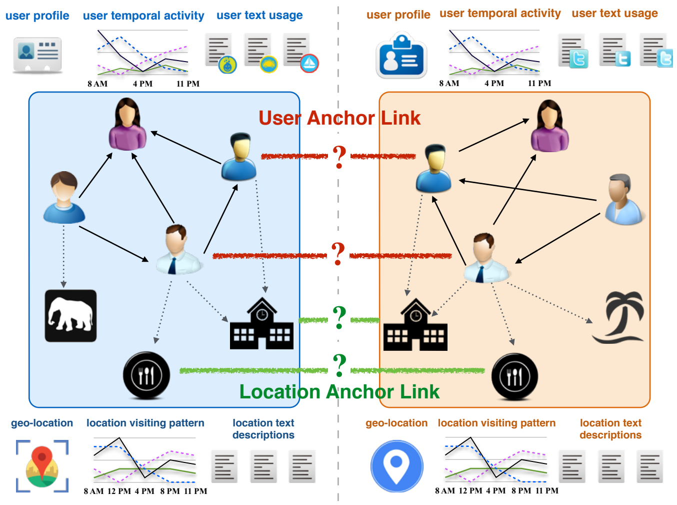


Figure 1: Example of the PCT problem.

explores various user attribute information only to match users between different social networks; (3) "*bipartite graph alignment*" [12], which aims at matching two bipartite graphs merely with the link information; and (4) "*homogeneous biological network matching*" [21], which studies the matching problem between two homogeneous PPI (protein-protein interaction) networks based on the structure information only.

As shown in Figure 1, different from all these related works, in the PCT problem, (1) few known anchor links connecting anchor instances between networks are available, and the lack of training instances will make the supervised methods [11] fail to work, (2) anchor instances contain heterogeneous information, including both attribute and link information (e.g., users have connections with other users, and also have attribute information, like profile information, temporal activity and text usage patterns; while locations have connections to users, and also have attribute information, e.g., geo-location, user visiting patterns and text descriptions), and (3) multiple different types of anchor links (i.e., user and location anchor links) are to be inferred simultaneously. More information about other related works is available in Section 5. To help illustrate the difference, we also summarize the differences of this paper from existing works in Table 1.

Despite its importance and novelty, the PCT problem is very challenging to solve due to:

- *heterogeneity of social networks*: anchor instances in online social networks can be associated with heterogeneous information, like various types of attributes and complex links. How to utilize such heterogeneous information to improve the network alignment results is very difficult.
- *unsupervised network co-alignment*: social network alignment with multiple types of anchor links has never been studied before and the PCT problem is still an open problem to this context so far. Furthermore, the unsupervised learning setting (due to the lack of known anchor links, i.e., training instances) poses extra challenges on addressing the PCT problem.
- *one-to-one property*: the anchor links to be inferred are assumed to have an inherent *one-to-one* constraint, i.e., each user/location can have at most one account in one network. (The case that users/locations have multiple accounts in one network can be addressed with [23], where duplicated accounts can be aggregated in advance to form one unique virtual account and the anchor links connecting these virtual accounts will still be "one-to-one".) How to preserve and utilize the constraint to improve network alignment results can be a great challenge.

To address the above challenges in PCT, a novel unsupervised network alignment framework UNICOAT (UNsupervIsed COncurrent AlignmenT) is proposed in this paper. Based on both attribute and link information, UNICOAT formulates the alignment problem as a joint optimization problem to infer both potential user and location anchor links. By relaxing the *one-to-one constraint* on anchor links, UNICOAT solves the optimization objective function with an alternative updating schema. Meanwhile, the introduced non-existing anchor links (by such a relaxation) can be further pruned with a minimum cost network flow based co-matching algorithm effectively.

The rest of the paper is organized as follows. We first introduce the terminology definitions and formulate the problem in Section 2. In Section 3, we propose the UNICOAT framework in detail. Section 4 presents the experiment results on real-world co-aligned social networks. Finally, in Sections 5-6, we describe the related works and conclude this paper.

2. PROBLEM FORMULATION

Before introducing the UNICOAT framework, we will give the definitions of some important concepts and formulation of the PCT problem first in this section.

2.1 Terminology Definition

Definition 1 (Attribute Augmented Social Network): Information entities (e.g., users and locations) in social networks studied in this paper can have both link and attribute information and such kind of networks can be formulated as *attribute augmented social networks*, $G = (\mathcal{V}, \mathcal{E}, \mathcal{B})$, where node set $\mathcal{V} = \mathcal{U} \cup \mathcal{L}$ contains both user and location nodes, link set $\mathcal{E} = \mathcal{E}_{u,u} \cup \mathcal{E}_{u,l}$ contains the links among users and those between users and locations. Attribute set $\mathcal{B} = \mathcal{B}_u \cup \mathcal{B}_l$, where $\mathcal{B}_u$ and $\mathcal{B}_l$ are the sets of attributes about users and locations respectively.

Definition 2 (Co-Aligned Attribute Augmented Social Networks): Social networks that share both common users and locations can be represented as *co-aligned attribute augmented social networks* $\mathcal{G} = ((G^{(1)}, G^{(2)}), (\mathcal{A}_u^{(1,2)}, \mathcal{A}_l^{(1,2)}))$, where $G^{(1)}$ and $G^{(2)}$ are two *attribute augmented social networks* respectively and $\mathcal{A}_u^{(1,2)}$, $\mathcal{A}_l^{(1,2)}$ are the sets of undirected user anchor links and location anchor links between networks $G^{(1)}$ and $G^{(2)}$ respectively. Link $(u^{(1)}, v^{(2)}) \in \mathcal{A}_u^{(1,2)}$ iff $u^{(1)}$ and $v^{(2)}$ are the accounts of the same user in $G^{(1)}$ and $G^{(2)}$ respectively, and similar for links in $\mathcal{A}_l^{(1,2)}$.

Traditional anchor links introduced in existing works [11, 34] normally represent the links connecting the same users' accounts in different networks (i.e., the *user anchor links* mentioned above). In this paper, we extend the definition of anchor links to any kinds of common information entities (e.g., users and locations) shared between networks and specify the definitions about user and location anchor links more clearly.

2.2 Problem Statement

Based on the above terminology definitions, we can present the PCT problem formally as follows.

Co-Alignment Problem: For any two given *attribute augmented social networks* $G^{(1)}$ and $G^{(2)}$, with the link and attribute information in both $G^{(1)}$ and $G^{(2)}$, the PCT problems aims at inferring

Table 1: Summary of related problems.

Property	PCT: Partial Co-Alignment	Anchor Link Inference [11]	User Matching across Networks [25]	Bipartite Network Alignment [3]	PPI Network Alignment [21]
network	heterogeneous	heterogeneous	heterogeneous	bipartite	homogeneous
information used	link&attribute	link&attribute	attribute	link	link
setting	unsupervised	supervised	supervised	unsupervised	unsupervised
# anchor links	multiple kinds	single kind	single kind	single kind	single kind

the potential anchor links between users and locations across $G^{(1)}$ and $G^{(2)}$ respectively. In other words, PCT explores the inference of both user anchor link and location anchor link sets $\mathcal{A}_u^{(1,2)}$ and $\mathcal{A}_l^{(1,2)}$ between $G^{(1)}$ and $G^{(2)}$ concurrently.

3. PROPOSED METHOD

In this section, we will introduce the UNICOAT framework to address the PCT problem in detail. Based on the link and attribute information, we will formulate the PCT problem as a joint optimization problem in Section 3.1 to infer potential user and location anchor links across networks. To solve the objective equation, we propose to relax the *one-to-one* constraint. And the non-existing redundant anchor links introduced by such a relaxation will be pruned with the network co-matching algorithm to be introduced in Section 3.2.

3.1 Anchor Links Co-Inference

As introduced in Section 2, let $\mathcal{A}_u^{(1,2)}$ be the set of inferred user anchor links between networks $G^{(1)}$ and $G^{(2)}$, which maps users between networks $G^{(1)}$ and $G^{(2)}$. Considering that users in different social networks are associated with both links and attribute information, the quality of the inferred anchor links $\mathcal{A}_u^{(1,2)}$ can be measured by the costs introduced by such mappings calculated with users' link and attribute information, i.e.,

$$cost(\mathcal{A}_u^{(1,2)}) = \text{ cost in links } (\mathcal{A}_u^{(1,2)}) + \alpha \cdot \text{ cost in attributes}(\mathcal{A}_u^{(1,2)}),$$

where α denotes the weight of the cost obtained from the attribute information (α is set as 1 in the experiments for simplicity, i.e., the link and attribute information is treated to be of the same importance). Considering that locations are also attached with link and attributes, similar cost function can be defined for the inferred location anchor links in $\mathcal{A}_l^{(1,2)}$:

$$cost(\mathcal{A}_l^{(1,2)}) = \text{ cost in links } (\mathcal{A}_l^{(1,2)}) + \alpha \cdot \text{ cost in attributes}(\mathcal{A}_l^{(1,2)}).$$

The optimal user and location anchor links $(\mathcal{A}_u^{(1,2)})^*$ and $(\mathcal{A}_l^{(1,2)})^*$ to be inferred in the PCT problem that can minimize the cost functions can be represented as

$$(\mathcal{A}_u^{(1,2)})^*, (\mathcal{A}_l^{(1,2)})^* = \arg \min_{\mathcal{A}_u^{(1,2)}, \mathcal{A}_l^{(1,2)}} cost(\mathcal{A}_u^{(1,2)}) + cost(\mathcal{A}_l^{(1,2)}).$$

To resolve the objective function, in the following parts of this section, we will introduce the (1) isolated user anchor link inference in subsection 3.1.1, (2) isolated location anchor link inference in subsection 3.1.2, and (3) the joint co-inference framework of user and location anchor links in subsection 3.1.3.

3.1.1 User Anchor Links Inference

Social connections among users clearly illustrate the social community structures of users in online social networks. Meanwhile, attribute information (e.g., profile information, text usage patterns, temporal activities) can reveal users' unique personal characteristics. Common users in different networks tend form similar community structures [32] and have very close personal characteristics [25]. As a result, link and attribute information about the users both plays very important roles in inferring potential user anchor links across networks. In this part, we will introduce how to use such information to improve the user anchor link inference results.

User Anchor Link Inference with Link Information

Based on the social links among users in both $G^{(1)}$ and $G^{(2)}$ (i.e., $\mathcal{E}_{u,u}^{(1)}$ and $\mathcal{E}_{u,u}^{(2)}$ respectively), we can construct the binary *social adjacency matrices* [17] $\mathbf{S}^{(1)} \in \mathbb{R}^{|\mathcal{U}^{(1)}| \times |\mathcal{U}^{(1)}|}$ and $\mathbf{S}^{(2)} \in \mathbb{R}^{|\mathcal{U}^{(2)}| \times |\mathcal{U}^{(2)}|}$ for networks $G^{(1)}$ and $G^{(2)}$ respectively. Entries in $\mathbf{S}^{(1)}$ and $\mathbf{S}^{(2)}$ (e.g., $\mathbf{S}^{(1)}(i,j)$ and $\mathbf{S}^{(2)}(l,m)$) will be assigned with value 1 iff the corresponding social links $(u_i^{(1)}, u_j^{(1)})$ and $(u_l^{(2)}, u_m^{(2)})$ exist in $G^{(1)}$ and $G^{(2)}$, where $u_i^{(1)}, u_j^{(1)} \in \mathcal{U}^{(1)}$ and $u_l^{(2)}, v_m^{(2)} \in \mathcal{U}^{(2)}$ are users in networks $G^{(1)}$ and $G^{(2)}$.

Via the inferred user anchor links $\mathcal{A}_u^{(1,2)}$, users as well as their social connections can be mapped between networks $G^{(1)}$ and $G^{(2)}$. We can represent the inferred user anchor links $\mathcal{A}_u^{(1,2)}$ with binary *user transitional matrix* $\mathbf{P} \in \mathbb{R}^{|\mathcal{U}^{(1)}| \times |\mathcal{U}^{(2)}|}$, where the (i_{th}, j_{th}) entry $\mathbf{P}(i,l) = 1$ iff link $(u_i^{(1)}, u_l^{(2)}) \in \mathcal{A}_u^{(1,2)}$. Considering that the constraint on user anchor links is *one-to-one*, each column and each row of $\mathbf{P}$ can contain at most one entry being assigned with value 1, i.e.,

$$\mathbf{P}\mathbf{1}^{|\mathcal{U}^{(2)}| \times 1} \leq \mathbf{1}^{|\mathcal{U}^{(1)}| \times 1}, \; \mathbf{P}^\top \mathbf{1}^{|\mathcal{U}^{(1)}| \times 1} \leq \mathbf{1}^{|\mathcal{U}^{(2)}| \times 1},$$

where $\mathbf{P}\mathbf{1}^{|\mathcal{U}^{(2)}| \times 1}$ and $\mathbf{P}^\top \mathbf{1}^{|\mathcal{U}^{(1)}| \times 1}$ can get the sum of rows and columns of matrix $\mathbf{P}$ respectively. Equation $\mathbf{P}\mathbf{1}^{|\mathcal{U}^{(2)}| \times 1} \leq \mathbf{1}^{|\mathcal{U}^{(1)}| \times 1}$ denotes that every entry of the left vector is no greater than the corresponding entry in the right vector.

Matrix $\mathbf{P}$ is an equivalent representation of user anchor link set $\mathcal{A}_u^{(1,2)}$. Next, we will infer the optimal *user transitional matrix* $\mathbf{P}$, from which we can obtain the optimal anchor link set $\mathcal{A}_u^{(1,2)}$.

The optimal user anchor links are those which can minimize the inconsistency of mapped social links across networks and the cost introduced by the inferred user anchor link set $\mathcal{A}_u^{(1,2)}$ with the link information can be represented as

$$\text{cost in link}(\mathcal{A}_u^{(1,2)}) = \text{cost in link}(\mathbf{P}) = \left\| \mathbf{P}^\top \mathbf{S}^{(1)} \mathbf{P} - \mathbf{S}^{(2)} \right\|_F^2,$$

where $\|\cdot\|_F$ denotes the Frobenius norm of the corresponding matrix and $\mathbf{P}^\top$ is the transpose of matrix $\mathbf{P}$.

User Anchor Link Inference with Attribute Information

Besides social links, users in social networks can be associated with a set of attributes, which can provide extra hints for identifying the correspondence relationships about users across networks. In this part, we will introduce the method to infer the user anchor links with attribute information, which includes *username information*, *text usage patterns* and *temporal activity information*.

Username that can differentiate users from each other in online social networks is like their online ID, which is an important factor in inferring potential anchor links. Let $(u_i^{(1)}, u_l^{(2)})$ be a potential anchor link between $G^{(1)}$ and $G^{(2)}$, the usernames of $u_i^{(1)}$ and $u_l^{(2)}$ can be represented as two sets of characters $n(u_i^{(1)})$ and $n(u_l^{(2)})$

respectively, based on which, various metrics proposed by Liu [25] can be applied to measure the similarity between $u_i^{(1)}$ and $u_l^{(2)}$. In this paper, we propose to calculate the similarity between the usernames with measure *Jaccard's Coefficient* [14], i.e.,

$$sim(n(u_i^{(1)}), n(u_l^{(2)})) = \frac{|n(u_i^{(1)}) \cap n(u_l^{(2)})|}{|n(u_i^{(1)}) \cup n(u_l^{(2)})|}.$$

Users usually have their unique active temporal patterns in online social networks [11]. For example, some users like to socialize with their online friends in the early morning, but some may prefer to do so in the evening after work. Users' online active time can be extracted based on their post publishing timestamps effectively. Let $\mathbf{t}(u_i^{(1)})$ and $\mathbf{t}(u_l^{(2)})$ be the normalized temporal activity distribution vectors of users $u_i^{(1)}$ and $u_l^{(2)}$, which are both of length 24. Entries of $\mathbf{t}(u_i^{(1)})$ and $\mathbf{t}(u_l^{(2)})$ contain the ratios of posts being published at the corresponding hour in a day. For example, $\mathbf{t}(u_i^{(1)})(3)$ denotes the ratio of all posts written by $u_1^{(1)}$ at 3AM. Based on vectors $\mathbf{t}(u_i^{(1)})$ and $\mathbf{t}(u_l^{(2)})$, we can calculate the inner product of the temporal distribution vectors [11] as the similarity scores between $u_i^{(1)}$ and $u_l^{(2)}$ in their temporal activity patterns, i.e.,

$$sim(\mathbf{t}(u_i^{(1)}), \mathbf{t}(u_l^{(2)})) = \mathbf{t}(u_i^{(1)})^\top \mathbf{t}(u_l^{(2)}).$$

Besides profile and online activity temporal distribution information, people normally have very different text usage habits online [25], which can reveal personal unique characteristics and can be applied in inferring the user anchor links across networks. We represent the text content used by users $u_i^{(1)}$ and $u_l^{(2)}$ as bag-of-words vectors [11], $\mathbf{w}(u_i^{(1)})$ and $\mathbf{w}(u_l^{(2)})$, weighted by TF-IDF [9] respectively. Commonly used text similarity measure: *Cosine similarity* [5] can be applied to measure the similarities in text usage patterns between $u_i^{(1)}$ and $u_l^{(2)}$, i.e.,

$$sim(\mathbf{w}(u_i^{(1)}), \mathbf{w}(u_l^{(2)})) = \frac{\mathbf{w}(u_i^{(1)})^\top \cdot \mathbf{w}(u_l^{(2)})}{\left\|\mathbf{w}(u_i^{(1)})\right\| \cdot \left\|\mathbf{w}(u_l^{(2)})\right\|}.$$

With these different attribute information (i.e., username, temporal activity and text content), we can calculate the similarities between users across networks $G^{(1)}$ and $G^{(2)}$. We represent such similarity matrix as $\mathbf{\Lambda} \in \mathbb{R}^{|\mathcal{U}^{(1)}| \times |\mathcal{U}^{(2)}|}$, where entry $\mathbf{\Lambda}(i,l)$ is the similarity between $u_i^{(1)}$ and $u_l^{(2)}$. $\mathbf{\Lambda}(i,l)$ can be represented as a combination of $sim(n(u_i^{(1)}), n(u_l^{(2)}))$, $sim(\mathbf{t}(u_i^{(1)}), \mathbf{t}(u_l^{(2)}))$ and $sim(\mathbf{w}(u_i^{(1)}), \mathbf{w}(u_l^{(2)}))$ and linear combination is used in in this paper due to its simplicity and wide usages. The optimal weights of similarity scores calculated with different attribute information can be learnt from the data theoretically, but it will make the model too complicated. To focus on the co-alignment problem itself, in this paper, we assume they are all of the same importance and propose to assign them with the same weight for simplicity concerns. In other words, $\mathbf{\Lambda}(i,l) = \frac{1}{3}\Big(sim(n(u_i^{(1)}), n(u_l^{(2)})) + sim(\mathbf{t}(u_i^{(1)}), \mathbf{t}(u_l^{(2)})) + sim(\mathbf{w}(u_i^{(1)}), \mathbf{w}(u_l^{(2)}))\Big)$.

Similar users across social networks are more likely to be the same user and user anchor links $\mathcal{A}_u^{(1,2)}$ that align similar users together should lead to lower cost. In this paper, the cost function introduced by the inferred user anchor links $\mathcal{A}_u^{(1,2)}$ in attribute information is represented as

$$\text{cost in attribute}(\mathcal{A}_u^{(1,2)}) = \text{cost in attribute}(\mathbf{P}) = -\left\|\mathbf{P} \circ \mathbf{\Lambda}\right\|_1,$$

where $\left\|\cdot\right\|_1$ is the L_1 norm [18] of the corresponding matrix, entry $(\mathbf{P} \circ \mathbf{\Lambda})(i,l)$ can be represented as $\mathbf{P}(i,l) \cdot \mathbf{\Lambda}(i,l)$ and $\mathbf{P} \circ \mathbf{\Lambda}$ denotes the Hadamard product [4] of matrices $\mathbf{P}$ and $\mathbf{\Lambda}$.

User Anchor Link Inference with Link and Attribute Information

Both link and attribute information is important for user anchor link inference. By taking these two categories of information into consideration simultaneously, we can represent the cost introduced by the inferred user anchor link set $\mathcal{A}_u^{(1,2)}$ as

$$\begin{aligned} cost(\mathcal{A}_u^{(1,2)}) &= \text{cost in link}(\mathcal{A}_u^{(1,2)}) + \alpha \cdot \text{cost in attribute}(\mathcal{A}_u^{(1,2)}) \\ &= \left\|\mathbf{P}^\top \mathbf{S}^{(1)} \mathbf{P} - \mathbf{S}^{(2)}\right\|_F^2 - \alpha \cdot \left\|\mathbf{P} \circ \mathbf{\Lambda}\right\|_1. \end{aligned}$$

The optimal *user transitional matrix* $\mathbf{P}^*$ which can lead to the minimum cost can be represented as

$$\begin{aligned} \mathbf{P}^* &= \arg\min_{\mathbf{P}} cost(\mathcal{A}_u^{(1,2)}) \\ &= \arg\min_{\mathbf{P}} \left\|\mathbf{P}^\top \mathbf{S}^{(1)} \mathbf{P} - \mathbf{S}^{(2)}\right\|_F^2 - \alpha \cdot \left\|\mathbf{P} \circ \mathbf{\Lambda}\right\|_1 \\ s.t. \quad & \mathbf{P} \in \{0,1\}^{|\mathcal{U}^{(1)}| \times |\mathcal{U}^{(2)}|}, \\ & \mathbf{P}\mathbf{1}^{|\mathcal{U}^{(2)}| \times 1} \leq \mathbf{1}^{|\mathcal{U}^{(1)}| \times 1}, \mathbf{P}^\top \mathbf{1}^{|\mathcal{U}^{(1)}| \times 1} \leq \mathbf{1}^{|\mathcal{U}^{(2)}| \times 1}. \end{aligned}$$

3.1.2 Location Anchor Links Inference

Similar to users, locations in online social networks are also associated with both link and attribute information (like the location links between users and locations, profile information and text descriptions about the locations, as well as the (longitude, latitude) coordinate information). The (longitude, latitude) pairs of the same location in different networks are usually not identical and various nearby locations can have very close coordinates, which pose great challenges in addressing the problem.

Location Anchor Link Inference with Link Information

Let $\mathcal{L}^{(1)}$ and $\mathcal{L}^{(2)}$ be the sets of locations in networks $G^{(1)}$ and $G^{(2)}$ respectively. Based on the location links between users and locations in networks $G^{(1)}$ and $G^{(2)}$ (i.e., $\mathcal{E}_{u,l}^{(1)}$ and $\mathcal{E}_{u,l}^{(2)}$), we can construct the binary *location adjacency matrices* $\mathbf{L}^{(1)} \in \mathbb{R}^{|\mathcal{U}^{(1)}| \times |\mathcal{L}^{(1)}|}$ and $\mathbf{L}^{(2)} \in \mathbb{R}^{|\mathcal{U}^{(2)}| \times |\mathcal{L}^{(2)}|}$ for networks $G^{(1)}$ and $G^{(2)}$ respectively. Entries in $\mathbf{L}^{(1)}$ and $\mathbf{L}^{(1)}$ $\big($e.g., $\mathbf{L}^{(1)}(i,j)$ and $\mathbf{L}^{(2)}(l,m)\big)$ are filled with value 1 iff user $u_i^{(1)}$ has visited location $l_j^{(1)}$ in $G^{(1)}$ and user $u_l^{(2)}$ has visited location $l_m^{(2)}$ in $G^{(2)}$.

Besides the *user transitional matrix* $\mathbf{P}$ which maps users between $G^{(1)}$ and $G^{(2)}$, we can also construct the binary *location transitional matrix* $\mathbf{Q} \in \{0,1\}^{|\mathcal{L}^{(1)}| \times |\mathcal{L}^{(2)}|}$ based on the inferred location anchor link set $\mathcal{A}_l^{(1,2)}$, which maps locations between $G^{(1)}$ and $G^{(2)}$. The cost introduced by the inferred location anchor link set $\mathcal{A}_l^{(1,2)}$ can be defined as the number of mis-mapped location links across networks, i.e.,

$$\text{cost in link}(\mathcal{A}_l^{(1,2)}) = \left\|\mathbf{P}^\top \mathbf{L}^{(1)} \mathbf{Q} - \mathbf{L}^{(2)}\right\|_F^2.$$

Location Anchor Link Inference with Attribute Information

In location-based social networks, each location has their own profile page, which shows the name and all the review comments about the location. Similar to the similarity scores for user anchor links, for any two locations $l_i \in \mathcal{L}^{(1)}$ and $l_m \in \mathcal{L}^{(2)}$, based on the names of locations l_i and l_m, we can calculate the similarity scores between l_i and l_m to be

$$sim(n(l_i), n(l_m)) = \frac{|n(l_i) \cap n(l_m)|}{|n(l_i) \cup n(l_m)|}.$$

Users' review comments can summarize the unique features about locations, which are also very important hints for inferring potential location anchor links. Similarly, we represent users' review comments posted as locations l_i and l_m as bag-of-words vectors weighted TF-IDF, $\mathbf{w}(l_i)$ and $\mathbf{w}(l_m)$. And the similarity between l_i and l_m based on the review comments can be represented as

$$sim(\mathbf{w}(l_i), \mathbf{w}(l_m)) = \mathbf{w}(l_i)^\top \cdot \mathbf{w}(l_i).$$

Closer locations are more likely to the same site than the ones which are far away. Based on the (latitude, longitude) information, we propose to define the similarity score between locations l_i and l_m as follows:

$$\begin{aligned}&sim((lat(l_i), long(l_i)), (lat(l_m), long(l_m))) = \\ &1.0 - \frac{\sqrt{(lat(l_i) - lat(l_m))^2 + (long(l_i) - long(l_m))^2}}{\sqrt{(180 - (-180))^2 + (90 - (-90))^2}}.\end{aligned}$$

Furthermore, we can also construct the similarity matrix between locations in $G^{(1)}$ and $G^{(2)}$ as $\mathbf{\Theta} \in \mathbb{R}^{|\mathcal{L}^{(1)}| \times |\mathcal{L}^{(2)}|}$, where entry $\mathbf{\Theta}(j, m) = \frac{1}{3}\Big(sim(n(l_i), n(l_m)) + sim(\mathbf{w}(l_i), \mathbf{w}(l_m)) + sim((lat(l_i), long(l_i)), (lat(l_m), long(l_i)))\Big)$. The optimal *location transitional matrix* $\mathbf{Q}$ which can minimize the cost in attribute information can be represented as

$$\text{cost in attribute}(\mathcal{A}_l^{(1,2)}) = - \left\|\mathbf{Q} \circ \mathbf{\Theta}\right\|_1.$$

Location Anchor Link Inference with Link and Attribute Information

By considering the location links and attributes attached to locations simultaneously, the cost function of inferred location anchor links $\mathcal{A}_l^{(1,2)}$ can be represented as

$$\begin{aligned}\text{cost}(\mathcal{A}_l^{(1,2)}) &= \text{cost in link}(\mathcal{A}_l^{(1,2)}) + \alpha \cdot \text{cost in attribute}(\mathcal{A}_l^{(1,2)}) \\ &= \left\|\mathbf{P}^\top \mathbf{L}^{(1)} \mathbf{Q} - \mathbf{L}^{(2)}\right\|_F^2 - \alpha \cdot \left\|\mathbf{Q} \circ \mathbf{\Theta}\right\|_1.\end{aligned}$$

The optimal user and location transitional matrices $\mathbf{P}^*$ and $\mathbf{Q}^*$ that can minimize the mapping cost will be

$$\begin{aligned}\mathbf{P}^*, \mathbf{Q}^* &= \arg\min_{\mathbf{P},\mathbf{Q}} \text{cost}(\mathcal{A}_l^{(1,2)}) \\ &= \arg\min_{\mathbf{P},\mathbf{Q}} \left\|\mathbf{P}^\top \mathbf{L}^{(1)} \mathbf{Q} - \mathbf{L}^{(2)}\right\|_F^2 - \alpha \cdot \left\|\mathbf{Q} \circ \mathbf{\Theta}\right\|_1, \\ s.t. \quad & \mathbf{Q} \in \{0,1\}^{|\mathcal{L}^{(1)}| \times |\mathcal{L}^{(2)}|}, \\ & \mathbf{Q}\mathbf{1}^{|\mathcal{L}^{(2)}| \times 1} \leq \mathbf{1}^{|\mathcal{L}^{(1)}| \times 1}, \mathbf{Q}^\top \mathbf{1}^{|\mathcal{L}^{(1)}| \times 1} \leq \mathbf{1}^{|\mathcal{L}^{(2)}| \times 1},\end{aligned}$$

where *location anchor links* also have *one-to-one* constraint, and the last two equations are added to maintain such a constraint.

3.1.3 Co-Inference of Anchor Links

User transitional matrix $\mathbf{P}$ is involved in the objective functions of inferring both *user anchor links* and *location anchor links*, and these two different anchor link inference tasks are strongly correlated (due to $\mathbf{P}$) and can be inferred simultaneously. By integrating the objective equations of anchor link inference for both users and locations, the optimal transitional matrices $\mathbf{P}^*$ and $\mathbf{Q}^*$ can be obtained simultaneously by solving the following objective function:

$$\begin{aligned}\mathbf{P}^*, \mathbf{Q}^* &= \arg\min_{\mathbf{P},\mathbf{Q}} \text{cost}(\mathcal{A}_u^{(1,2)}) + \text{cost}(\mathcal{A}_l^{(1,2)}) \\ &= \arg\min_{\mathbf{P},\mathbf{Q}} \left\|\mathbf{P}^\top \mathbf{S}^{(1)} \mathbf{P} - \mathbf{S}^{(2)}\right\|_F^2 + \left\|\mathbf{P}^\top \mathbf{L}^{(1)} \mathbf{Q} - \mathbf{L}^{(2)}\right\|_F^2 \\ &\quad - \alpha \cdot \left\|\mathbf{P} \circ \mathbf{\Lambda}\right\|_1 - \alpha \cdot \left\|\mathbf{Q} \circ \mathbf{\Theta}\right\|_1, \\ s.t. \quad & \mathbf{P} \in \{0,1\}^{|\mathcal{U}^{(1)}| \times |\mathcal{U}^{(2)}|}, \mathbf{Q} \in \{0,1\}^{|\mathcal{L}^{(1)}| \times |\mathcal{L}^{(2)}|}, \\ & \mathbf{P}\mathbf{1}^{|\mathcal{U}^{(2)}| \times 1} \leq \mathbf{1}^{|\mathcal{U}^{(1)}| \times 1}, \mathbf{P}^\top \mathbf{1}^{|\mathcal{U}^{(1)}| \times 1} \leq \mathbf{1}^{|\mathcal{U}^{(2)}| \times 1}, \\ & \mathbf{Q}\mathbf{1}^{|\mathcal{L}^{(2)}| \times 1} \leq \mathbf{1}^{|\mathcal{L}^{(1)}| \times 1}, \mathbf{Q}^\top \mathbf{1}^{|\mathcal{L}^{(1)}| \times 1} \leq \mathbf{1}^{|\mathcal{L}^{(2)}| \times 1}.\end{aligned}$$

The objective function is an constrained $0-1$ integer programming problem, which is hard to address mathematically. Many relaxation algorithms have been proposed so far [1]. To solve the problem, in this paper, we propose to relax the binary constraint of matrices $\mathbf{P}$ and $\mathbf{Q}$ to real numbers in range $[0,1]$ and entries in $\mathbf{P}$ and $\mathbf{Q}$ will denote the existence probabilities/confidence scores of the corresponding anchor links. Redundant anchor links introduced by such a relaxation will be pruned with the co-matching algorithm to be introduced in the next section.

Meanwhile, the Hadamard product terms $\mathbf{P} \circ \mathbf{\Lambda}$ and $\mathbf{Q} \circ \mathbf{\Theta}$ can be very hard to deal with when solving the optimization problem. Considering that matrices $\mathbf{P}$, $\mathbf{\Lambda}$, $\mathbf{Q}$ and $\mathbf{\Theta}$ are all positive matrices, we will replace the L_1 norm of Hadamard product terms with the following Lemmas.

Lemma 1: For any given matrix $\mathbf{A}$, the square of its Frobenius norm equals to the trace of $\mathbf{A}\mathbf{A}^\top$, i.e., $\left\|\mathbf{A}\right\|_F^2 = tr(\mathbf{A}\mathbf{A}^\top)$.

Lemma 2: For two given positive matrices $\mathbf{A}$ and $\mathbf{B}$ of the same dimensions, the L_1 norm of the Hadamard product about $\mathbf{A}$ and $\mathbf{B}$ equals to the trace of $\mathbf{A}^\top\mathbf{B}$ or $\mathbf{A}\mathbf{B}^\top$, i.e., $\left\|\mathbf{A} \circ \mathbf{B}\right\|_1 = tr(\mathbf{A}^\top\mathbf{B}) = tr(\mathbf{A}\mathbf{B}^\top)$.

PROOF. According to the definitions of matrix trace, terms $tr(\mathbf{A}^\top\mathbf{B})$ and $tr(\mathbf{A}\mathbf{B}^\top)$ equals to the Frobenius product [18] of matrices $\mathbf{A}$ and $\mathbf{B}$, i.e.,

$$tr(\mathbf{A}^\top\mathbf{B}) = tr(\mathbf{A}\mathbf{B}^\top) = \sum_{i,j} \mathbf{A}(i,j)\mathbf{B}(i,j).$$

Meanwhile,

$$\left\|\mathbf{A} \circ \mathbf{B}\right\|_1 = \sum_{i,j} \left|(\mathbf{A} \circ \mathbf{B})(i,j)\right| = \sum_{i,j} \left|\mathbf{A}(i,j) \cdot \mathbf{B}(i,j)\right|.$$

Considering that both $\mathbf{A}$ and $\mathbf{B}$ are positive matrices, so the following equation can always hold:

$$\left\|\mathbf{A} \circ \mathbf{B}\right\|_1 = \sum_{i,j} \mathbf{A}(i,j) \cdot \mathbf{B}(i,j) = tr(\mathbf{A}^\top\mathbf{B}) = tr(\mathbf{A}\mathbf{B}^\top).$$

□

To solve the objective function, in this paper, we will follow the Alternating Projected Gradient Descent (APGD) method introduced in [12] and the *one-to-one* constraint is relaxed, where constraints $\mathbf{P}\mathbf{1} \leq \mathbf{1}$, $\mathbf{P}^\top\mathbf{1} \leq \mathbf{1}$ will be replaced with $\left\|\mathbf{P}\right\|_1 \leq t$ instead, where t is a small constant. Similarly, the *one-to-one* constraint on $\mathbf{Q}$ is also relaxed and replaced with $\left\|\mathbf{Q}\right\|_1 \leq t$. Furthermore, by incorporating terms $\left\|\mathbf{P}\right\|_1$ and $\left\|\mathbf{Q}\right\|_1$ into the minimization objective function. Based on the relaxed constraints as well as

Lemmas 1-2, the new objective function can be represented to be

$$\begin{aligned}\arg\min_{\mathbf{P},\mathbf{Q}}\; f(\mathbf{P},\mathbf{Q}) &= tr\Big((\mathbf{P}^\top\mathbf{S}^{(1)}\mathbf{P}-\mathbf{S}^{(2)})(\mathbf{P}^\top\mathbf{S}^{(1)}\mathbf{P}-\mathbf{S}^{(2)})^\top\Big)\\ &+ tr\Big((\mathbf{P}^\top\mathbf{L}^{(1)}\mathbf{Q}-\mathbf{L}^{(2)})(\mathbf{P}^\top\mathbf{L}^{(1)}\mathbf{Q}-\mathbf{L}^{(2)})^\top\Big)\\ &- \alpha\cdot tr(\mathbf{P}\mathbf{\Lambda}^\top) - \alpha\cdot tr(\mathbf{Q}\mathbf{\Theta}^\top) + \gamma\cdot\|\mathbf{P}\|_1 + \mu\cdot\|\mathbf{Q}\|_1\\ s.t.\quad &\mathbf{0}^{|\mathcal{U}^{(1)}|\times|\mathcal{U}^{(2)}|} \le \mathbf{P} \le \mathbf{1}^{|\mathcal{U}^{(1)}|\times|\mathcal{U}^{(2)}|},\\ &\mathbf{0}^{|\mathcal{L}^{(1)}|\times|\mathcal{L}^{(2)}|} \le \mathbf{Q} \le \mathbf{1}^{|\mathcal{L}^{(1)}|\times|\mathcal{L}^{(2)}|},\end{aligned}$$

where γ and μ denote the weights on $\|\mathbf{P}\|_1$ and $\|\mathbf{Q}\|_1$ respectively.

As we can see, the objective function is with respect to $\mathbf{P}$ and $\mathbf{Q}$ and we cannot give a closed-form solution for the objective function. In this paper, we propose to calculate the optimal $\mathbf{P}$ and $\mathbf{Q}$ with alternative updating procedure based on the gradient descent algorithm: (1) fix $\mathbf{Q}$ and minimize the objective function *w.r.t.* $\mathbf{P}$; and (2) fix $\mathbf{P}$ and minimize the objective function *w.r.t.* $\mathbf{Q}$. If during these two updating procedures, entries in $\mathbf{P}$ or $\mathbf{Q}$ become invalid, we use a projection to guarantee the $[0,1]$ constraint: (1) if $\mathbf{P}(i,j) > 1$ or $\mathbf{Q}(i,j) > 1$, we project it to 1; and (2) if $\mathbf{P}(i,j) < 0$ or $\mathbf{Q}(i,j) < 0$, we project it to 0 [12]. Matrices $\mathbf{P}$ and $\mathbf{Q}$ can be initialized with the method introduced in the Experiment Setting Section, and the alternative updating equations of these two matrices are available as follow:

$$\begin{aligned}\mathbf{P}^\tau &= \mathbf{P}^{\tau-1} - \eta_1\cdot\frac{\partial\Gamma(\mathbf{P}^{\tau-1},\mathbf{Q}^{\tau-1},\gamma,\mu)}{\partial\mathbf{P}}\\ &= \mathbf{P}^{\tau-1} - 2\eta_1\cdot\Big(\mathbf{S}^{(1)}\mathbf{P}\mathbf{P}^\top(\mathbf{S}^{(1)})^\top\mathbf{P} + (\mathbf{S}^{(1)})^\top\mathbf{P}\mathbf{P}^\top\mathbf{S}^{(1)}\mathbf{P}\\ &+ \mathbf{L}^{(1)}\mathbf{Q}\mathbf{Q}^\top(\mathbf{L}^{(1)})^\top\mathbf{P} - \mathbf{S}^{(1)}\mathbf{P}(\mathbf{S}^{(2)})^\top - (\mathbf{S}^{(1)})^\top\mathbf{P}\mathbf{S}^{(2)}\\ &- \mathbf{L}^{(1)}\mathbf{Q}(\mathbf{L}^{(2)})^\top - \frac{1}{2}\alpha\mathbf{\Lambda} + \frac{1}{2}\gamma\mathbf{1}\mathbf{1}^\top\Big),\end{aligned}$$

$$\begin{aligned}\mathbf{Q}^\tau &= \mathbf{Q}^{\tau-1} - \eta_2\cdot\frac{\partial\Gamma(\mathbf{P}^{\tau},\mathbf{Q}^{\tau-1},\gamma,\mu)}{\partial\mathbf{Q}}\\ &= \mathbf{Q}^{\tau-1} - 2\eta_2\cdot\Big((\mathbf{L}^{(1)})^\top\mathbf{P}\mathbf{P}^\top\mathbf{L}^{(1)}\mathbf{Q} - (\mathbf{L}^{(1)})^\top\mathbf{P}\mathbf{L}^{(2)}\\ &- \frac{1}{2}\alpha\mathbf{\Theta} + \frac{1}{2}\mu\mathbf{1}\mathbf{1}^\top\Big),\end{aligned}$$

where η_1 and η_2 are the *search steps* in updating $\mathbf{P}$ and $\mathbf{Q}$ respectively. Such a updating process will continue until both $\mathbf{P}$ and $\mathbf{Q}$ converge. The optimal learning rates η_1 and η_2 obtaining the minimum $f(\mathbf{P}^\tau,\mathbf{Q}^\tau)$ can be represented as

$$\begin{aligned}\eta_1^{(\tau)} &= \arg_{\eta_1}\min f(\mathbf{P}^\tau,\mathbf{Q}^\tau),\\ \eta_2^{(\tau)} &= \arg_{\eta_2}\min f(\mathbf{P}^\tau,\mathbf{Q}^\tau).\end{aligned}$$

The functions can be addressed by taking derivative of $f(\cdot)$ with regards to η_1 (or η_2) and make it equal to 0, we can obtain a cubic equation involving η_1 (or η_2). Multiple roots may exist when addressing the equation and the representation of the roots is very complicated. In this paper, for simplicity, we propose to assign η_1 and η_2 with a constant value (i.e., 0.05 in the experiments).

3.2 Network Flow based Co-Matching

To solve the objective function, the *one-to-one* constraints on both *user anchor links* and *location anchor links* are relaxed, which can take values in range $[0,1]$. As a result, users and locations in each network can be connected by multiple user/location anchor links of various confidence scores across networks simultaneously and the *one-to-one* constraint can no longer hold any more. To maintain such a constraint on both user and location anchor links, we propose to prune the redundant ones introduced due to the relaxation with *network flow* based network co-matching algorithm in this subsection.

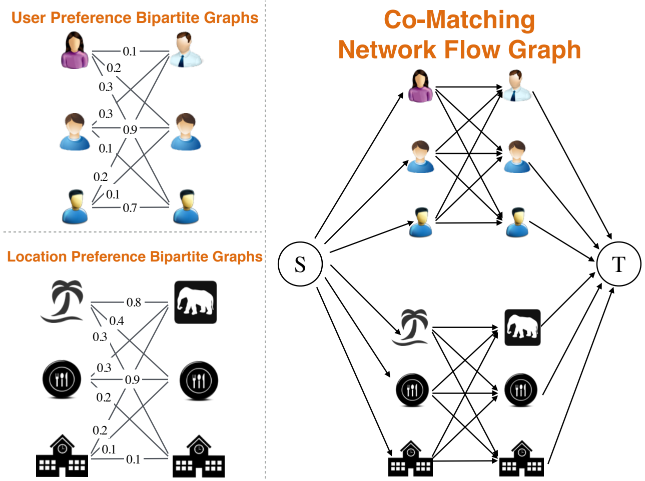

User Preference Bipartite Graphs
0.1
0.2
0.3
0.3
0.9
0.1
0.2
0.1
0.7
Location Preference Bipartite Graphs
0.8
0.4
0.3
0.3
0.9
0.2
0.2
0.1
0.1
Co-Matching Network Flow Graph
S
T

Figure 2: User and Location Preference Bipartite Graphs and Co-Matching Network Flow Graph.

Based on user sets $\mathcal{U}^{(1)}$ and $\mathcal{U}^{(2)}$, location sets $\mathcal{L}^{(1)}$ and $\mathcal{L}^{(2)}$, as well as the existence confidence scores of potential user and location anchor links between networks $G^{(1)}$ and $G^{(1)}$ (i.e., entries of $\mathbf{P}$ and $\mathbf{Q}$), we can construct the user and location preference bipartite graphs as shown in the left plots of Figure 2.

User Preference Bipartite Graph

The *user preference bipartite graph* can be represented as $BG_{\mathcal{U}} = (\mathcal{U}^{(1)}\cup\mathcal{U}^{(2)}, \mathcal{U}^{(1)}\times\mathcal{U}^{(2)}, \mathcal{W}_{\mathcal{U}})$, where $\mathcal{U}^{(1)}\cup\mathcal{U}^{(2)}$ denotes the user nodes in $G^{(1)}$ and $G^{(2)}$, $\mathcal{U}^{(1)}\times\mathcal{U}^{(2)}$ contains all the potential user anchor links between $G^{(1)}$ and $G^{(2)}$, and $\mathcal{W}_{\mathcal{U}}$ will map links in $\mathcal{U}^{(1)}\times\mathcal{U}^{(2)}$ to their confidence scores (i.e., entries in $\mathbf{P}$) inferred in the previous section.

Location Preference Bipartite Graph

Similarly, we can also represent the *location preference bipartite graph* to be $BG_{\mathcal{L}} = (\mathcal{L}^{(1)}\cup\mathcal{L}^{(2)}, \mathcal{L}^{(1)}\times\mathcal{L}^{(2)}, \mathcal{W}_{\mathcal{L}})$, where the weight mapping of potential *location anchor links* (i.e., $\mathcal{W}_{\mathcal{L}}$) can be obtained from *location transitional matrix* $\mathbf{Q}$ in a similar way as introduced before.

Co-Matching Network Flow Graph

In this paper, we employ traditional network flow algorithm to match users and locations across networks $G^{(1)}$ and $G^{(2)}$ simultaneously, which are grouped together in an integrated network flow model, named "co-matching network flow". As shown in the right plot of Figure 2, based on the *user preference bipartite graphs* and *location preference bipartite graphs*, we propose to construct the *co-matching network flow graph* by adding (1) a source node S, (2) a sink node T, (3) links connecting node S and links in $\mathcal{U}^{(1)}\cup\mathcal{L}^{(1)}$ (i.e., $\{S\}\times(\mathcal{U}^{(1)}\cup\mathcal{L}^{(1)})$), and (4) links connecting nodes in $\mathcal{U}^{(2)}\cup\mathcal{L}^{(2)}$ and node T (i.e., $(\mathcal{U}^{(2)}\cup\mathcal{L}^{(2)})\times\{T\}$).

Bound Constraint

In the network flow model, each link in the *co-matching network flow graph* is associated with a *upper bound* and *lower bound* to control the amount of flow going through it. For example, the upper and lower bounds of potential user anchor link $(u,v)\in\mathcal{U}^{(1)}\times\mathcal{U}^{(2)}$ in the *co-matching network flow graph* can be represented as

$$\underline{B}(u,v) \le F(u,v) \le \overline{B}(u,v),$$

where $F(u,v)$ denotes the flow amount going through link (u,v), $\underline{B}(u,v)$ and $\overline{B}(u,v)$ represent the *lower bound* and *upper bound* associated with link (u,v) respectively.

Considering that the constraint on both user and location anchor links is *one-to-one* and networks studied in this paper are partially aligned, users in online social networks include both anchor and non-anchor users; so is the case for locations. In other words, each user and location in online social networks can be connected by at most one anchor links across networks, which can be achieved by adding the following upper and lower bound constraint on links $\{S\} \times (\mathcal{U}^{(1)} \cup \mathcal{L}^{(1)})$ and $(\mathcal{U}^{(2)} \cup \mathcal{L}^{(2)}) \times \{T\}$:

$$0 \leq F(u,v) \leq 1, \forall (u,v) \in \{S\} \times (\mathcal{U}^{(1)} \cup \mathcal{L}^{(1)}) \cup (\mathcal{U}^{(2)} \cup \mathcal{L}^{(2)}) \times \{T\}.$$

Among all the potential user anchor links in $\mathcal{U}^{(1)} \times \mathcal{U}^{(2)}$ and location anchor links in $\mathcal{L}^{(1)} \times \mathcal{L}^{(2)}$, only part of these links will be selected finally due to the *one-to-one* constraint. To represent whether a link (u,v) is selected or not, we set the flow amount going through links $\mathcal{U}^{(1)} \times \mathcal{U}^{(2)} \cup \mathcal{L}^{(1)} \times \mathcal{L}^{(2)}$ as integers with upper and lower bounds to be 0 an 1 (1 denotes the link is selected, and 0 otherwise) respectively, i.e.,

$$F(u,v) \in \{0,1\}, \forall (u,v) \in \mathcal{U}^{(1)} \times \mathcal{U}^{(2)} \cup \mathcal{L}^{(1)} \times \mathcal{L}^{(2)}.$$

Mass Balance Constraint

In addition, in network flow model, for each node in the graph (except the source and sink node), the amount of flow going through it should meet the *mass balance constraint*, i.e., for each node in the network, the amount of network flow going into it should equals to that going out from it:

$$\sum_{w \in \mathcal{N}_F, (w,u) \in \mathcal{L}_F} F(w,u) = \sum_{v \in \mathcal{N}_F, (u,v) \in \mathcal{L}_F} F(u,v),$$

where $\mathcal{N}_F = \{S\} \cup \mathcal{U}^{(1)} \cup \mathcal{U}^{(2)} \cup \mathcal{L}^{(1)} \cup \mathcal{L}^{(2)} \cup \{T\}$ denotes all the nodes in the *co-matching network flow graph* and $\mathcal{L}_F = \{S\} \times (\mathcal{U}^{(1)} \cup \mathcal{L}^{(1)}) \cup \mathcal{U}^{(1)} \times \mathcal{U}^{(2)} \cup \mathcal{L}^{(1)} \times \mathcal{L}^{(2)} \cup (\mathcal{U}^{(2)} \cup \mathcal{L}^{(2)}) \times \{T\}$ represents all the links in graph.

Maximum Confidence Objective Function

All the potential links connecting users and locations across networks are associated with certain costs in network flow model, where links with lower costs are more likely to be selected. In this paper, we modify the model a little and aim at selecting the links introducing the maximum confidence scores instead from $\mathcal{U}^{(1)} \times \mathcal{U}^{(2)}$ and $\mathcal{L}^{(1)} \times \mathcal{L}^{(2)}$ respectively, which can be obtained with the following objective functions:

$$\max \sum_{(u,v) \in (\mathcal{U}^{(1)} \times \mathcal{U}^{(2)})} F(u,v) \cdot \mathcal{W}_{\mathcal{U}}(u,v).$$

$$\max \sum_{(m,n) \in (\mathcal{L}^{(1)} \times \mathcal{L}^{(2)})} F(m,n) \cdot \mathcal{W}_{\mathcal{L}}(m,n).$$

The final objective equation of simultaneous *co-matching* of users and locations across networks can be represented to be

$$\begin{aligned}
\max \quad & \sum_{(u,v) \in (\mathcal{U}^{(1)} \times \mathcal{U}^{(2)})} F(u,v) \cdot \mathcal{W}_{\mathcal{U}}(u,v) + \\
& \sum_{(m,n) \in (\mathcal{L}^{(1)} \times \mathcal{L}^{(2)})} F(m,n) \cdot \mathcal{W}_{\mathcal{L}}(m,n), \\
s.t. \quad & 0 \leq F(u,v) \leq 1, \forall (u,v) \in \{S\} \times (\mathcal{U}^{(1)} \cup \mathcal{L}^{(1)}) \cup (\mathcal{U}^{(2)} \cup \mathcal{L}^{(2)}) \times \{T\}, \\
& F(u,v) \in \{0,1\}, \forall (u,v) \in \mathcal{U}^{(1)} \times \mathcal{U}^{(2)} \cup \mathcal{L}^{(1)} \times \mathcal{L}^{(2)}, \\
& \sum_{w \in \mathcal{N}_F, (w,u) \in \mathcal{L}_F} F(w,u) = \sum_{v \in \mathcal{N}_F, (u,v) \in \mathcal{L}_F} F(u,v).
\end{aligned}$$

The above network flow objective function can be solved with open-source toolkits (e.g., Scipy.Optimization[4] and GLPK[5]) and the detailed derivative steps will not be introduced here due to the limited space. In the obtained solution, the flow amount variable of potential user and location anchor links achieving value 1 are the selected ones which will be assigned with label $+1$, while the remaining (i.e., those achieving value 0) are not selected which are assigned with label -1.

Table 2: Properties of the Heterogeneous Networks

		network	
	property	**Twitter**	**Foursquare**
# node	user	5,223	5,392
	tweet/tip	9,490,707	48,756
	location	297,182	38,921
# link	friend/follow	164,920	76,972
	write	9,490,707	48,756
	locate	615,515	48,756

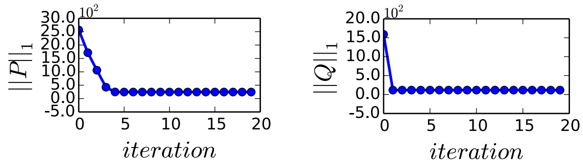


Figure 3: Convergence analysis of iterative updating method

4. EXPERIMENTS

To test the effectiveness of the proposed UNICOAT model, in this section, extensive experiments will be done on two real-world partially co-aligned online social networks: Foursquare and Twitter. We will describe the datasets used in this paper at first and then introduce the experiment settings in detail. Finally, we will show the experiment results and give brief analysis about the results.

4.1 Dataset Descriptions

The social networks dataset used in this paper are Foursquare and Twitter, which are co-aligned by both users and locations shared between these two networks. These two social network datasets are crawled during November, 2012, whose statistical information is available in Table 2. More detailed descriptions and the crawling method is available in [28, 34].

4.2 Experiment Settings

In this part, we will introduce the experiment settings in detail, which include (1) comparison methods, (2) evaluation metrics, and (3) experiment setups.

4.2.1 Comparison Methods

To show the advantages of UNICOAT in addressing the PCT problem, we compare UNICOAT with many different baseline methods. Considering that no known user and location anchor links are available actually in the PCT problem, as a result, no existing supervised network alignment methods (e.g., MNA [11]) can be applied. All the comparison methods are based on unsupervised learning settings, which can be divided into 4 categories:

Co-Alignment Methods

- UNICOAT: Method UNICOAT introduced in this paper can align two online social networks based on the shared users and locations simultaneously, which consists of two steps:

[4] http://docs.scipy.org/doc/scipy/reference/optimize.html

[5] http://www.gnu.org/software/glpk/

(1) unsupervised potential user and location anchor links inference; (2) co-matching of social networks to prune redundant anchor links to maintain the *one-to-one* constraint.

Bipartite Graph Alignment Methods

- BigAlign: Method BigAlign is a bipartite network alignment methods introduced in [12], which can align two bipartite graphs (e.g., user-product bipartite graph) simultaneously with link information only.
- BigAlignExt: Method BigAlignExt is a bipartite network alignment methods introduced in this paper. BigAlignExt can align user-location bipartite networks with both location links between users and locations as well as attribute information about users and locations across networks.

Isolated Alignment Methods

- ISO: Method ISO is an unsupervised network alignment method introduced in [12]. ISO merely infers the user anchor links only based on the friendship information among users.
- ISOExt: Method ISOExt is an unsupervised network alignment method proposed in this paper, which is identical to ISO but utilizes both friendship links among users and attribute information of users.

Traditional Unsupervised Link Prediction Methods

- Relative Degree Distance based Network Alignment: RDD is the heuristics based unsupervised network alignment method introduced in [12] to fill in the initial values of the cross-network transitional matrices, e.g., $\mathbf{P}$ and $\mathbf{Q}$ in this paper. For any two users/location $u_l^{(i)}$ and $u_m^{(j)}$ in networks $G^{(i)}$ and $G^{(j)}$, the relative degree distance between them can be represented as $RDD(u_l^{(i)}, u_m^{(j)}) = \left(1 + \frac{|deg(u_l^{(i)}) - deg(u_m^{(j)})|}{(deg(u_l^{(i)}) + deg(u_m^{(j)}))/2}\right)^{-1}$. High relative degree distance denotes lower confidence score of anchor link $(u_l^{(i)}, u_m^{(j)})$.

Table 3: Performance comparison of different methods for inferring user anchor links (uniCOAT here denotes the first step of uniCOAT only).

measure		θ				
	methods	1	2	3	4	5
AUC	uniCOAT	**0.868**	**0.831**	**0.814**	**0.804**	**0.799**
	BigAlignExt	0.813	0.779	0.759	0.752	0.749
	BigAlign	0.568	0.557	0.555	0.552	0.550
	ISOExt	0.818	0.782	0.762	0.754	0.61
	ISO	0.547	0.529	0.52	0.518	0.516
	RDD	0.531	0.530	0.523	0.514	0.508
Prec@100	uniCOAT	**0.705**	**0.688**	**0.657**	**0.640**	**0.556**
	BigAlignExt	0.587	0.507	0.472	0.434	0.327
	BigAlign	0.347	0.284	0.265	0.228	0.220
	ISOExt	0.427	0.391	0.373	0.352	0.301
	ISO	0.301	0.253	0.225	0.216	0.208
	RDD	0.234	0.228	0.207	0.172	0.127

Table 4: Performance comparison of different methods for inferring location anchor links (uniCOAT here denotes the first step of uniCOAT only).

measure		θ				
	methods	1	2	3	4	5
AUC	uniCOAT	**0.822**	**0.815**	**0.796**	**0.794**	**0.753**
	BigAlignExt	0.698	0.695	0.672	0.667	0.662
	BigAlign	0.592	0.586	0.576	0.572	0.56
	RDD	0.54	0.526	0.52	0.506	0.504
Prec@100	uniCOAT	**0.695**	**0.658**	**0.636**	**0.610**	**0.535**
	BigAlignExt	0.507	0.434	0.372	0.328	0.327
	BigAlign	0.407	0.325	0.293	0.284	0.275
	RDD	0.216	0.204	0.183	0.182	0.157

4.2.2 Evaluation Metrics

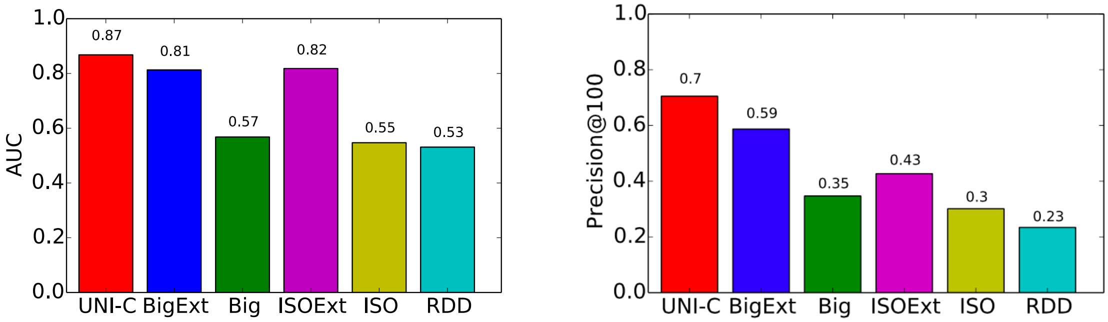
Methods uniCOAT (the first step), BigAlign, BigAlignExt ISO, ISOExt and RDD can output the confidence scores of potential inferred links but no labels are available, whose performance can be evaluated by metrics like AUC and Precision@100, etc. As to method uniCOAT, links selected finally in the matching are assumed to achieve confidence score 1.0 and label $+1$, while the remaining can achieve confidence score 0.0 and label -1. As a result, uniCOAT can also output the labels of potential anchor links, whose performance can be evaluated by various metrics, e.g., AUC, Precision@100, Precision, Recall, F1 and Accuracy simultaneously.

4.2.3 Experiment Setup

In the experiments, all the known user anchor links and location anchor links are used for evaluation only, which are not used in building models at all. Initially, a fully co-aligned Foursquare and Twitter involving 200 users and 200 locations are randomly sampled from the data. To obtain networks of different partial alignment degrees, extra non-anchor users and locations are added to the network controlled by partial alignment rate $\theta = \frac{\text{\#total item}}{\text{\#anchor item}} \in \{1, 2, 3, 4, 5\}$, where $\theta = 1$ denote full alignment and $\theta = 5$ means $\frac{\text{\#total item}}{\text{\#anchor item}} = 5$, i.e., extra 800 non-anchor users and non-anchor locations are added to the network. We first calculate the social adjacency matrices $\mathbf{S}^{(1)}$, $\mathbf{S}^{(2)}$ and location adjacency matrices $\mathbf{L}^{(1)}$, $\mathbf{L}^{(2)}$ based on the social links among users and location links between users and locations. With the attribute information, we can represent the user similarity matrix as $\mathbf{\Lambda}$ and location similarity matrix as $\mathbf{\Theta}$ respectively. Parameter α is set as 1 in the experiments for simplicity. Before co-updating the user and location transitional matrices $\mathbf{P}$ and $\mathbf{Q}$, entries in $\mathbf{P}$ and $\mathbf{Q}$ are initialized with the *relative degree distance* scores between users and locations across networks. Matrices $\mathbf{P}$ and $\mathbf{Q}$ will be updated with equations given in Section 3.1.3 until convergence. The values of learning rates η_1 and η_2 are set as constant 0.05 in the experiments. Based on the updated matrices $\mathbf{P}$ and $\mathbf{Q}$, we can get the scores of potential user anchor links and location anchor links across networks and further prune the non-existing ones with the network co-matching method introduced in Section 3.2. Links selected finally are labeled as $+1$ links with confidence 1.0 (to be real anchor links) and the remaining are labeled as -1 links with confidence 0 (to be non-existing anchor links) instead.

4.3 Convergence Analysis

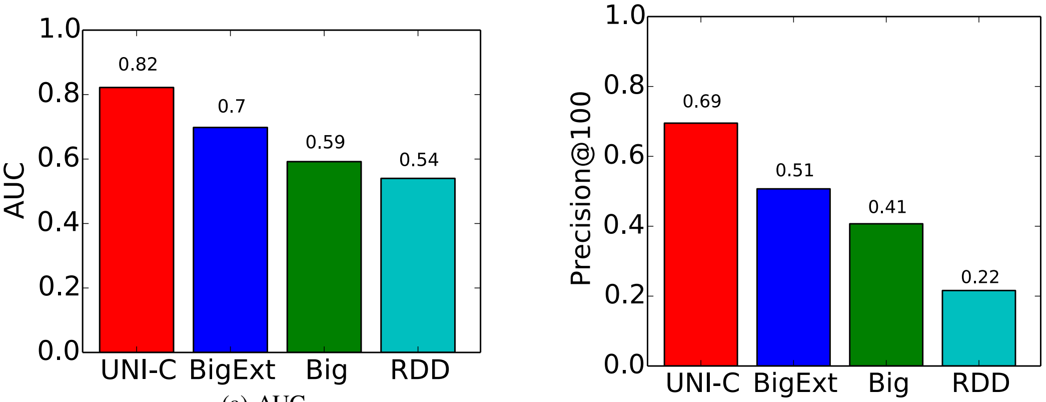
To solve the objective function, we propose to update matrices $\mathbf{P}$ and $\mathbf{Q}$ iteratively until convergence. To show that with the co-



(a) AUC

(b) Prec@100

Figure 4: Performance of methods without matching in inferring user anchor links (UNICOAT here denotes the first step of UNICOAT only).



(a) AUC

(b) Prec@100

Figure 5: Performance of methods without matching in inferring location anchor links (UNICOAT here denotes the first step of UNICOAT only).

updating equations, matrices $\mathbf{P}$ and $\mathbf{Q}$ can do converge, we show the L_1 norm of matrices $\mathbf{P}$ and $\mathbf{Q}$ in each iteration is shown in Figure 3, where parameter θ is set as 1 (i.e., the networks are fully co-aligned and all the users and locations are anchor instances). As shown in the figures, as the mutual updating continues, the L_1 norm of both $\mathbf{P}$ and $\mathbf{Q}$ can converge very quickly to around 200 in less than 5 iterations.

4.4 Experiment Results

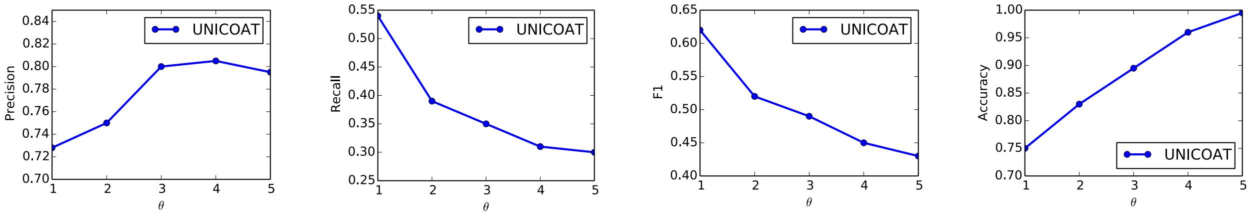
The experiment results of addressing the PCT problem are available in Tables 3-4 and Figures 4-7.

In Figure 4 and 5, we fix $\theta = 1$ and show the results achieved by comparison methods without matching step (i.e., methods UNICOAT (the first step), BIGALIGN, BIGALIGNEXT, ISO, ISOEXT and RDD) evaluated by AUC and Precision@100. Methods ISO and ISOEXT can only be applied to align networks via user generated information, which are not compared in the alignment results of locations (i.e., Figure 5). In both Figure 4 and 5, we can observe that (1) UNICOAT performs the best among all the comparison methods in inferring user and location anchor links evaluated by both AUC and Precision@100. For example, in Figure 4, UNICOAT can achieve AUC score of 0.87, which is over 6% better than BIGALIGNEXT and ISOEXT, and 50% higher than the AUC score achieved by BIGALIGN, ISO and RDD. Similar performance of UNICOAT is available in other plots. It demonstrates that utilizing the heterogeneous information in the network to infer user and location anchor links simultaneously can improve the results a lot. (2) BIGALIGNEXT and ISOEXT can achieve better performance than BIGALIGN and ISO. Recalling that methods BIGALIGNEXT and ISOEXT use both the link and attribute information, while BIGALIGN and ISO use the link information. It justifies that the attribute information of both users and locations is helpful for inferring anchor links across networks. (3) By comparing UNICOAT with RDD (i.e., the initialization method of matrices $\mathbf{P}$ and $\mathbf{Q}$ in UNICOAT), we observe that UNICOAT can outperform RDD with significant advantages. It proves the effectiveness of the proposed network co-alignment model, which can obtain better results than the initial value.

4.5 Sensitivity Analysis

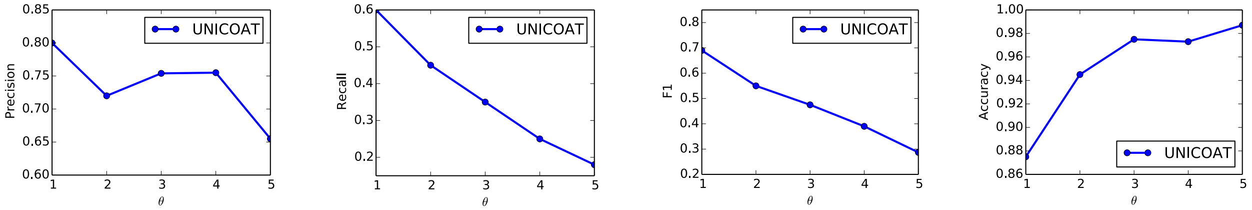
In Figures 4-5, parameter θ is fixed as 1. In Tables 3-4, we further change it with values in $\{1, 2, 3, 4, 5\}$ by adding more non-anchor users and locations into the network. Generally, with more non-anchor users and locations, the PCT will become more difficult and the performance of all the methods will degrade, but UNICOAT can achieve the best performance consistently. For example, when $\theta = 5$, the AUC score achieved by UNICOAT in inferring social links is 0.799, which is 6.7%, 45%, 31%, 54.8% and 57.2% higher than that gained by BIGALIGNEXT, BIGALIGN, ISOEXT, ISO and RDD respectively. Similar observations can be obtained from the user anchor links inference results evaluated by Precision@100, and location anchor link inference by both AUC and Precision@100 in Tables 3-4.

In the previous part, we have shown the performance of methods without matching step, while anchor links inferred by which cannot meet the *one-to-one* constraint. Next, we will test the ef-



(a) Precision (b) Recall (c) F1 (d) Accuracy

Figure 6: Performance of methods with matching in inferring user anchor links (UNICOAT here includes both two steps of UNICOAT).



(a) Precision (b) Recall (c) F1 (d) Accuracy

Figure 7: Performance of methods with matching in inferring location anchor links (UNICOAT here includes both two steps of UNICOAT).

fectiveness of the matching step in pruning the non-existing anchor links and the results achieved by UNICOAT (the second step) are shown in Figures 6-7. Parameter θ are assigned with values in $\{1, 2, 3, 4, 5\}$. The anchor links inferred by UNICOAT can all meet the one-to-one constraint and are of high quality. For example, when $\theta = 1$, the Precision, Recall, F1 and Accuracy achieved by UNICOAT are 0.73, 0.54, 0.62 and 0.75 respectively in inferring user anchor links. As θ increases, Recall and F1 scores achieved by UNICOAT will decrease as it will be more hard to identify the real anchor links among larger number of potential ones. Meanwhile, the Precision and Accuracy of UNICOAT will increase. The potential reason can be due to the class imbalance problem. By adding more non-anchor users to the network, more non-existing anchor links (i.e., the negative class links) will be introduced and UNICOAT can achieve higher Precision and Accuracy by predicting more negative instances correctly.

5. RELATED WORKS

Network alignment problem is an important research problem, which have been studied in various areas, e.g., protein-protein-interaction network alignment in bioinformatics [10, 13, 20], chemical compound matching in chemistry [22], data schemas matching data warehouse [16], ontology alignment web semantics [7], graph matching in combinatorial mathematics [15], and figure matching and merging in computer vision [6, 2].

In recent years, witnessing the rapid growth of online social networks, researchers start to shift their attention to align multiple online social networks. Homogeneous network alignment was studied in [24], enlightened by which the problem of aligning two bipartite networks is studied by Koutra [12], where a fast alignment algorithm which can be applied to large-scale networks is introduced. Users can have various types of attribute information in social networks generated by their social activities, based on which Zafarani et al. study the cross-network user matching problem in [25]. In addition to attribute information, Kong et al. [11] propose to fully align social networks with the heterogeneous link and attribute information simultaneously based on a supervised learning setting. Besides fully aligning different social networks, Zhang et al. propose a framework for partial social network alignment in [29], where the constraint on anchor links is "one-to-one$_{\leq}$". Anchor links are very hard to obtain and to make use of the small amount known anchor links, Zhang et al. formulate the network alignment as a PU learning problem instead [31]. In addition, users nowadays are usually involved in more than two social networks, a general multiple (more than two) network alignment framework is introduced in [33], which utilize the "transitivity law" property of anchor links to identify the optimal results.

Across the aligned networks, various application problems have been studied. Cross-site heterogeneous link prediction problems are studied by Zhang et al. [28, 27, 34, 31] by transferring links across partially aligned networks. Besides link prediction problems, Jin and Zhang et al. proposes to partition multiple large-scale social networks simultaneously in [30, 32, 8]. The problem of information diffusion across partially aligned networks is studied by Zhan et al. in [26], where the traditional LT diffusion model is extended to the multiple heterogeneous information setting. Shi et al. give a comprehensive survey about the existing works on heterogeneous information networks in [19], which includes a section talking about network information fusion works and related application problems in detail.

6. CONCLUSION

Multiple kinds of information entities can be shared across networks, e.g., users and locations. In this paper, simultaneously inference of the anchor links connecting common users and common locations across heterogeneous networks is studied. A novel unsupervised co-alignment framework UNICOAT is introduced in this paper, which consists of two phrases: (1) co-inference of potential user and location anchor links based on an unsupervised learning setting, and (2) co-matching of networks to prune non-existing anchor links and maintain the *one-to-one* constraint on anchor links. Extensive experiments conducted on real-world social network datasets demonstrate the outstanding performance of UNICOAT.

7. ACKNOWLEDGEMENT

This work is supported in part by NSF through grants III-1526499, CNS-1115234, and OISE-1129076, and Google Research Award.

8. REFERENCES

[1] Y. Aflaloa, A. Bronsteinb, and R. Kimmel. On convex relaxation of graph isomorphism. In *PNAS*, 2015.

[2] M. Bayati, M. Gerritsen, D. Gleich, A. Saberi, and Y. Wang. Algorithms for large, sparse network alignment problems. In *ICDM*, 2009.

[3] I. Bhattacharya and L. Getoor. Collective entity resolution in relational data. *TKDD*, 2007.

[4] D. Chandler. The norm of the schur product operation. *Numerische Mathematik*, 1962.

[5] M. Charikar. Similarity estimation techniques from rounding algorithms. In *STOC*, 2002.

[6] D. Conte, P. Foggia, C. Sansone, and M. Vento. Thirty years of graph matching in pattern recognition. *IJPRAI*, 2004.

[7] A. Doan, J. Madhavan, P. Domingos, and A. Halevy. Ontology matching: A machine learning approach. In *Handbook on Ontologies*. 2004.

[8] S. Jin, J. Zhang, P. Yu, S. Yang, and A. Li. Synergistic partitioning in multiple large scale social networks. In *IEEE BigData*, 2014.

[9] T. Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In *ICML*, 1997.

[10] M. Kalaev, V. Bafna, and R. Sharan. Fast and accurate alignment of multiple protein networks. In *RECOMB*. 2008.

[11] X. Kong, J. Zhang, and P. Yu. Inferring anchor links across multiple heterogeneous social networks. In *CIKM*, 2013.

[12] D. Koutra, H. Tong, and D. Lubensky. Big-align: Fast bipartite graph alignment. In *ICDM*, 2013.

[13] C. Liao, K. Lu, M. Baym, R. Singh, and B. Berger. Isorankn: spectral methods for global alignment of multiple protein networks. *Bioinformatics*, 2009.

[14] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM*, 2003.

[15] F. Manne and M. Halappanavar. New effective multithreaded matching algorithms. In *IPDPS*, 2014.

[16] S. Melnik, H. Garcia-Molina, and E. Rahm. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *ICDE*, 2002.

[17] M. Newman. Analysis of weighted networks. *Physical Review E*, 2004.

[18] K. Petersen and M. Pedersen. The matrix cookbook. Technical report, 2012.

[19] C. Shi, Y. Li, J. Zhang, Y. Sun, and P. Yu. A survey of heterogeneous information network analysis. *CoRR*, 2015.

[20] R. Singh, J. Xu, and B. Berger. Pairwise global alignment of protein interaction networks by matching neighborhood topology. In *RECOMB*, 2007.

[21] R. Singh, J. Xu, and B. Berger. Global alignment of multiple protein interaction networks with application to functional orthology detection. *Proceedings of the National Academy of Sciences*, 2008.

[22] A. Smalter, J. Huan, and G. Lushington. Gpm: A graph pattern matching kernel with diffusion for chemical compound classification. In *IEEE BIBE*, 2008.

[23] M. Tsikerdekis and S. Zeadally. Multiple account identity deception detection in social media using nonverbal behavior. *IEEE TIFS*, 2014.

[24] S. Umeyama. An eigendecomposition approach to weighted graph matching problems. *IEEE TPAMI*, 1988.

[25] R. Zafarani and H. Liu. Connecting users across social media sites: A behavioral-modeling approach. In *KDD*, 2013.

[26] Q. Zhan, S. Wang J. Zhang, P. Yu, and J. Xie. Influence maximization across partially aligned heterogenous social networks. In *PAKDD*, 2015.

[27] J. Zhang, X. Kong, and P. Yu. Predicting social links for new users across aligned heterogeneous social networks. In *ICDM*, 2013.

[28] J. Zhang, X. Kong, and P. Yu. Transfer heterogeneous links across location-based social networks. In *WSDM*, 2014.

[29] J. Zhang, W. Shao, S. Wang, X. Kong, and P. Yu. Pna: Partial network alignment with generic stable matching. In *IRI*, 2015.

[30] J. Zhang and P. Yu. Community detection for emerging networks. In *SDM*, 2015.

[31] J. Zhang and P. Yu. Integrated anchor and social link predictions across partially aligned social networks. In *IJCAI*, 2015.

[32] J. Zhang and P. Yu. Mcd: Mutual clustering across multiple heterogeneous networks. In *IEEE BigData Congress*, 2015.

[33] J. Zhang and P. Yu. Multiple anonymized social networks alignment. In *ICDM*, 2015.

[34] J. Zhang, P. Yu, and Z. Zhou. Meta-path based multi-network collective link prediction. In *KDD*, 2014.