


User identity linkage across social networks via linked heterogeneous network embedding

Yaqing Wang¹  · Chunyan Feng¹ · Ling Chen² ·
Hongzhi Yin³ · Caili Guo⁴ · Yunfei Chu⁴

Received: 30 November 2017 / Revised: 27 February 2018 / Accepted: 11 April 2018 /
Published online: 23 April 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract User identity linkage has important implications in many cross-network applications, such as user profile modeling, recommendation and link prediction across social networks. To discover accurate cross-network user correspondences, it is a critical prerequisite to find effective user representations. While structural and content information describe users from different perspectives, there is a correlation between the two aspects of

This article belongs to the Topical Collection: *Special Issue on Web and Big Data*
Guest Editors: Junjie Yao, Bin Cui, Christian S. Jensen, and Zhe Zhao

✉ Yaqing Wang
wangyq@bupt.edu.cn

Chunyan Feng
cyfeng@bupt.edu.cn

Ling Chen
ling.chen@uts.edu.au

Hongzhi Yin
db.hongzhi@gmail.com

Caili Guo
guocaili@bupt.edu.cn

Yunfei Chu
yfchu@bupt.edu.cn

- ¹ Beijing Key Laboratory of Network System Architecture and Convergence, School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China
- ² Center for Artificial Intelligence, University of Technology Sydney, Sydney, Australia
- ³ School of Information Technology and Electrical Engineering, University of Queensland, Brisbane, Australia
- ⁴ Beijing Laboratory of Advanced Information Networks, Beijing, China

information. For example, a user who follows a celebrity tends to post content about the celebrity as well. Therefore, the projections of structural and content information of a user should be as close to each other as possible, which inspires us to fuse the two aspects of information in a unified space. However, owing to the information heterogeneity, most existing methods extract features from content and structural information respectively, instead of describing them in a unified way. In this paper, we propose a Linked Heterogeneous Network Embedding model (LHNE) to learn the comprehensive representations of users by collectively leveraging structural and content information in a unified framework. We first model the topics of user interests from content information to filter out noise. Next, cross-network structural and content information are embedded into a unified space by jointly capturing the friend-based and interest-based user co-occurrence in intra-network and inter-network, respectively. Meanwhile, LHNE learns user transfer and topic transfer for enhancing information exchange across networks. Empirical results show LHNE outperforms the state-of-the-art methods on both real social network and synthetic datasets and can work well even with little or no structural information.

Keywords User identity linkage · Network embedding · Transfer learning · Heterogeneous social network

1 Introduction

In recent years, due to the popular and diverse functionalities of social networks, more and more users simultaneously own accounts on multiple social networks such as Twitter, Flickr, or Instagram [21]. Linking user accounts in different social networks has very important influence in many cross-network applications. For user profile modeling [4, 11], a comprehensive understanding of a user's interests can be obtained by aggregating the user's historical behaviors in different networks. For cross-network recommendation [12, 27, 28] and link prediction [6, 32–34], anchor users (i.e., identity linked users) mitigate the cold start and data sparsity problems by enabling information transferring between aligned networks. However, because of the unrevealing nature of the Web and the fact that most social network platforms preserve the anonymity of users, the correspondences among users' different accounts are also unrevealed. Therefore, an interesting question arises - how can we find user correspondences in different social networks?

Most of the existing research on user identity linkage [3, 7, 9, 31] focuses on extracting user characteristics from user contributed content information (e.g., blogs or tweets posted by users) and structural information (e.g., connections and interactions between users), by assuming the independence between the two types of information. Hence, existing studies usually handle the content information and the structural information separately. However, while the two types of information describe users from different perspectives, we note that there exist correlations between structural and content information. For example, it is very likely for a Twitter user to post tweets about a celebrity if s/he follows (likes) the celebrity. Therefore, it is expected that the structural information and content information may share a common space where the structure-based and content-based representations of users are close to each other. We are thus motivated to model user characteristics by fusing structural and content information in a unified way in linking user identities in different social networks.

It is, however, a challenging task to fuse structural and content information to learn the effective representations of users, because of the following two main issues. Firstly, content

and structural information come from heterogeneous feature spaces (such as different granularities and data structures), which makes it hard to fuse them in a joint space. Secondly, the information are diverse and noisy across different networks. For example, users may upload images as content in one social network (e.g., Flickr) and post textual content in another (e.g., Twitter). The diversity problem makes it extremely difficult to leverage the diverse types of information simultaneously to accurately link user identities. Also, user content is noisy with information irrelevant to characterize user identities, such as advertisements. Similarly, the network structure may be noisy as well, because not all edges represent true “friend” relations [21].

Recently, as representing a network as low-dimensional vectors is an efficient way to solve high computation and space cost problem [2], methods that embed multiple types of information of a network into a low-dimensional space have attracted a great deal of attention in a variety of fields, such as text mining and recommendation. For instance, Tang et al. [23] proposed a text embedding method based on modeled heterogeneous text networks, which is proved to be useful for document classification. Xie et al. [25] proposed a generic graph-based embedding model, which jointly captures the sequential effect, geographical influence, temporal cyclic effect and semantic effect in a unified way for the recommendation task. However, these methods are either applied to individual networks or not designed for user identity linkage. There are also some studies [10, 13] focusing on aligning users across social networks by network embedding. Nevertheless, these methods consider only structural information represented as homogeneous networks. Hence, to achieve user linkage with higher performance, we design an effective method that jointly embeds structural and content information in multiple heterogeneous networks.

In this paper, we propose a Linked Heterogeneous Network Embedding model (LHNE) to learn the comprehensive descriptions of users in different social networks through jointly leveraging structural and content information in a unified framework. First, we model the topics of user interests to represent the content information in different social networks at a same granularity and filter out the noise. Second, we capture *friend-based* (i.e., structure) and *interest-based* (i.e., content) user co-occurrence in linked heterogeneous network using four types of sub-networks (i.e., user-user intra/inter-network and user-topic intra/inter-network). Third, we learn the effective user representations by embedding the sub-networks into a unified low-dimensional space. In the meantime, to bridge different social networks, we learn user transfer and topic transfer using a set of seed users. Finally, users are mapped by computing the similarity between the representations of users in different networks.

The main contributions are summarized as follows.

1. We focus on learning the comprehensive representations of users by jointly leveraging structural and content information in a unified way, and integrating network structures and content into linked heterogeneous network, which incorporates the friend-based and interest-based user co-occurrence in different social networks.
2. We propose a novel network embedding model “LHNE”, which embeds the linked heterogeneous network into a unified low-dimensional space in terms of intra-network and inter-network. In the meantime, we learn user and topic transfer across social networks to solve the diversity problem utilizing a set of seed users as prior information.
3. We demonstrate the performance of LHNE on both real social network and synthetic datasets. A series of experimental results validate that LHNE achieves better performance than the state-of-the-art methods in terms of effectiveness, reliability and sensibility and can work well even with little or no structural information.

The remainder of the paper is organized as follows. Section 2 reviews existing work related to our research. Section 3 defines concepts and terms used in this paper and formally defines the user identity linkage problem. Section 4 details the technology of our proposed LHNE model. Experimental results on both real social network and synthetic datasets are presented in Section 5. We conclude our work in Section 6.

2 Related work

There are many studies addressing the user identity linkage problem by exploring a variety types of user information in multiple social networks, including profile information, structural information and content information. We group existing methods for user identity linkage into the following two main categories.

The first category of methods exploits one type of user information for user identity linkage. The most intuitive way is to use profile information [15, 30], such as username, avatar and gender. However, profile information contains many null and inconsistent values, which makes it very hard to achieve satisfactory linkage accuracy. For the purpose of performance enhancing, many studies leverage structural information [8, 10, 13, 16, 22] or content information [18, 20] to discover user correspondences in different social networks. The common idea shared by structure-based methods is to extract neighborhood-based features as the inputs of models. For example, Narayanan et al. [16] proposed a graph theoretic model based on the number of common neighbors to perform user identity linkage task. Korula et al. [8] designed a parallelizable mapping algorithm based on neighborhood-based features such as the degrees of unmapped users and the number of common neighbors. Moreover, to solve the high-dimensional problem of networks, techniques are employed to embed networks into a low-dimensional space, which is followed by effective representation learning for users to link user identities [10, 13]. For instance, Liu et al. [10] proposed a network representation learning method to simultaneously learn the follower-ship/followee-ship of individual users, and used seed users as constraints for user representation learning across networks. Man et al. [13] presented a supervised framework that learns embedding-based representations of nodes and links user's accounts by a projection method. Meanwhile, many studies have shown that content information is also conducive for user identity linkage. Phan et al. [18] regarded the user identity linkage task as a pairwise classification problem based on the content browsed by users on different devices, and then used the gradient boosting method to detect same users. Riederer et al. [20] utilized an aligning algorithm to compute affinity scores based on time-stamped location data and then adopted a maximum weighted matching scheme to find the most likely candidate pair. Overall, exploiting only one specific type of information leads to incomplete and biased user features, which impairs the performance of user identity linkage.

The second category of methods aims to harness multiple types of user information to improve the accuracy of user identity linkage [3, 7, 9, 31]. Kong et al. [7] developed a SVM classifier with one-to-one constraint to predict anchor links by integrating neighborhood-based network features and content features. Zhang et al. [31] proposed a unified link prediction framework for collective link identification (inter-links and intra-links), which also extracts features from both structural and content information [7]. Cao et al. [3] adopted a bootstrapping method, which respectively extracts features from usernames, social ties and content, and then learns model parameters by the EM algorithm. Liu et al. [9] proposed a multi-objective framework by modeling heterogeneous behaviors (e.g., profile features

and content features) and structure consistency, respectively. However, most of existing methods extract user features from different types of information separately, and then combine them together as model inputs. Since features extracted from different information sources have different feature spaces and underlying interpretations, it may not be ideal to directly concatenating them as input features.

Our work in this paper distinguishes itself from other research in the following three aspects.

1. Unlike most prior works on anchor link prediction [3, 7, 31] and user identity linkage [9] that assume the independence between content and structural information, our model aims to jointly leverage structural and content information in a unified framework.
2. Although several studies [10, 13] have exploited network embedding methods for user identity linkage, they are all based on structural information that are represented as homogeneous networks. In contrast, we propose a novel network embedding method to improve linkage performance based on heterogeneous networks including both structural and content information.
3. For the purpose of enhancing information exchange across networks, we solve the cross-network diversity problem by learning user transfer and topic transfer across social networks using a set of seed users.

3 Problem definition

In this section, we define preliminary concepts used in this paper and the user identity linkage problem. Without loss of generality, we focus on user identity linkage in two social networks, while the settings of two social networks can be easily extended to multiple networks.

Let $G = \{U, E\}$ be a social network, where U is the set of users and $E = U \times U$ is the set of edges in G representing the social connections between users. Each user $u \in U$ is associated with a vector of words $V_u = \{v_1, v_2, \dots, v_n\}$ representing the content contributed by the user u in G . Each edge $e_{ij} \in E$, connecting users u_i and u_j , is associated with a weight w_{ij} , denoting the correlations between u_i and u_j . For example, if G is a co-authorship network, w_{ij} is the number of times u_i and u_j have co-authored.

Then, the problem of user identity linkage in two social networks can be formally defined as follows.

Problem 1. (User Identity Linkage) Given two social networks $G^x = \{U^x, E^x\}$ and $G^y = \{U^y, E^y\}$, the task of user identity linkage is to predict whether a pair of users $u_i^x \in U^x$ and $u_j^y \in U^y$ belong to a same real natural person.

4 Linked heterogeneous network embedding model

In this section, we first model the topics of user interests with content information, and present the LHNE method mathematically based on friend-based and interest-based proximity of users in terms of intra-network embedding (intra-NE) and inter-network embedding (inter-NE). Then, we present the joint learning of user embedding and topic embedding of different networks in a unified low-dimensional space. Next, we map users across social

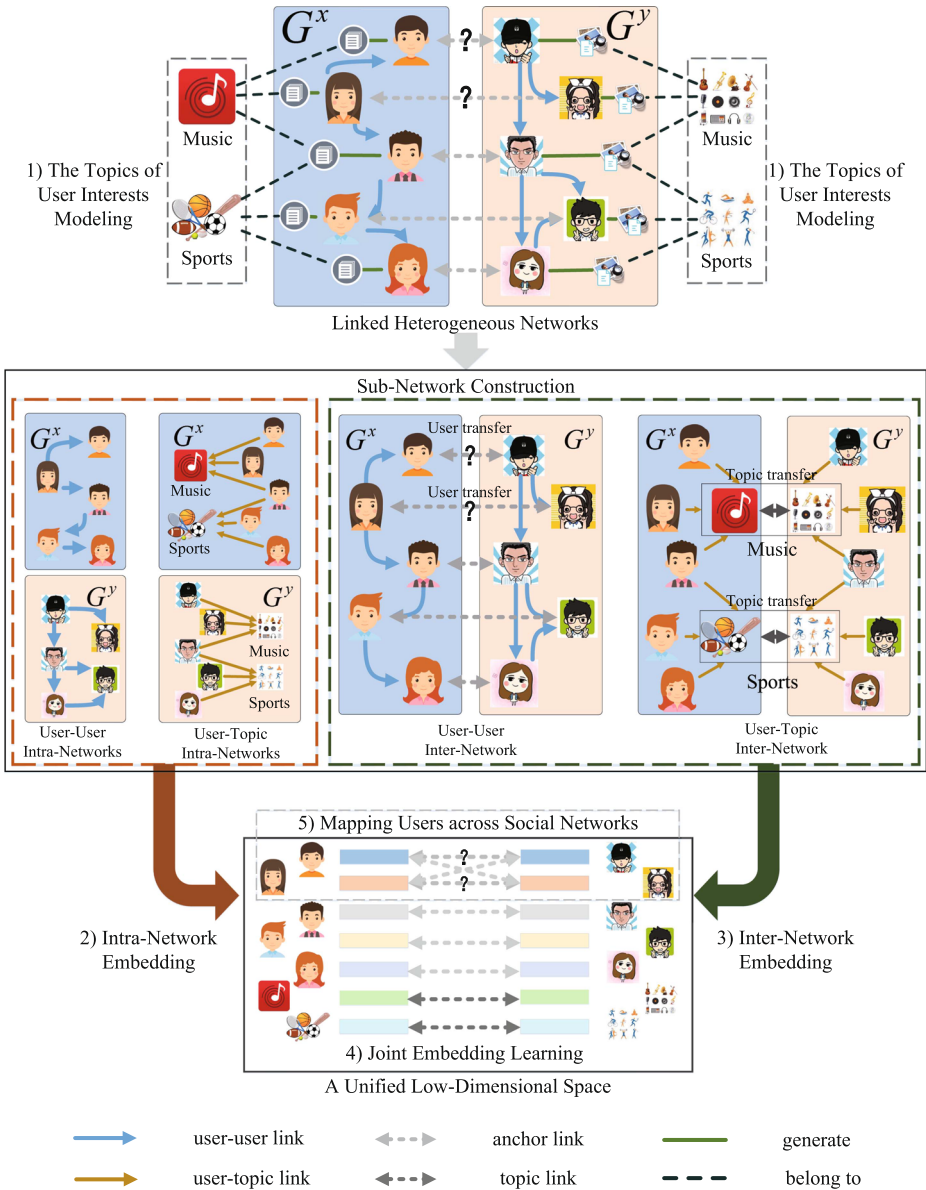


Figure 1 Illustration of the LHNE framework

networks based on the representations of users. The illustration of LHNE is depicted in Figure 1. Finally, we present the pseudo codes to summarize the overall algorithm.

4.1 The topics of user interests modeling

As discussed in Section 2, user content information is not only diverse but also noisy. In order to exploit effectively user content information, we capture content-wise user proximity

by modeling the topics of user interests from the content contributed by users in a social network.

In particular, given a social network G , we adopt Latent Dirichlet Allocation (LDA) [1] to model topics from the set of word vectors associated with users $V = \{V_u | u \in U\}$. After obtaining the topic distribution from V via LDA, we can capture user proximities through the topics of user interests.

The detailed distributions for LDA model are as below:

$$\theta_i \sim \text{Dir}(\alpha), \phi_k \sim \text{Dir}(\beta), z_{ij} \sim \text{Multi}(\theta_i), w_{ij} \sim \text{Multi}(\phi_{z_{ij}}) \quad (1)$$

where α, β are hyper-parameters. $\text{Dir}(\cdot)$ is the Dirichlet distribution and $\text{Multi}(\cdot)$ is the Multinomial distribution. For each user i , we draw his topic distribution θ_i from the Dirichlet distribution with the parameter α ($\theta_i \sim \text{Dir}(\alpha)$). For each word, we first draw the topic z_{ij} from user's topic distribution ($z_{ij} \sim \text{Multi}(\theta_i)$), and then select the word w_{ij} according to topic-word dictionary $\phi_{z_{ij}}$. The topic-word dictionary ϕ_k also follows the Dirichlet distribution with the parameter β ($\phi_k \sim \text{Dir}(\beta)$).

We train LDA model by estimating the model parameters with the Gibbs sampling method. We can derive the Gibbs updating rule as follows:

$$P(z_{ij} = k | z_{-ij}, w, \phi, \cdot) \propto \frac{n_{i,k}^{-ij} + \alpha_k}{\sum_{p=1}^P (n_{i,p}^{-ij} + \alpha_p)} + \frac{n_{k,j}^{-ij} + \beta_j}{\sum_{q=1}^Q (n_{k,q}^{-ij} + \beta_q)} \quad (2)$$

where $n_{i,k}$ is the number of times topic k being assigned to user i (number of times $z_i = k$) and $n_{k,j}$ is the number of times word j being assigned to topic k . After sufficient sampling iterations, the topic distribution θ_i can be estimated by:

$$\hat{\theta}_{i,k} = \frac{n_{i,k} + \alpha_k}{\sum_{p=1}^P (n_{i,p} + \alpha_p)} \quad (3)$$

In our experiments, we notice that some topics are not important to capture user proximities in terms of interests. Therefore, instead of taking into account the complete set of topics, we select the set $T_i = \{k | \theta_{i,k} > h\}$, where h is the topic threshold. By setting a suitable threshold h , we can improve not only the computation efficiency with a reduced number of topics but also the robustness by filtering noises represented by insignificant topics.

Note that, we model topics of user interests from individual social networks, instead of collectively extracting topics from the two social networks. The reason is that, the content information of social networks is noisy and diverse. It is expected that topics modeled from individual social networks will be of high quality than those extracted from combined social networks. It will also allow more flexibility in parameter setting. For example, we may set different topic numbers for different social networks to make the topics semantically meaningful.

4.2 Intra-network embedding

Given a social network G , we first apply intra-network embedding to embed it into a low-dimensional space by preserving friend-based proximities and interest-based proximities within a network. In particular, we perform the following two tasks.

Task 1. (User-User Intra-NE) The target is to preserve the friend-based proximities of users within a network. The intuitive idea is to make the representations of users sharing common neighbors to be as similar as possible.

Task 2. (User-Topic Intra-NE) The target is to preserve the interest-based proximities of users within a network. That is, the representations of users who are interested in same topics are expected to be similar.

The first task can be performed directly on the a given social network, e.g., $G = (U, E)$. For the second task, we construct a user-topic bipartite network, defined as $G_{ut} = \{U \cup T, E_{ut}\}$, where U is the set of users, T is the set of topics extracted by LDA from the content contributed by users, and E_{ut} is the set of edges connecting users and topics. Each edge connecting user u_i with topic t_j is associated with a weight w_{ij} representing the probabilities u_i is interested in topic t_j , which can be obtained from the output of the LDA model (i.e., (3)).

Then, for both tasks 1 and 2, similar to existing representation learning methods [23], we first define the conditional probability between two nodes v_i and v_j as follows,

$$p(v_j|v_i) = \frac{\exp(\mathbf{z}_j \cdot \mathbf{z}_i)}{\sum_{v_k \in V_B} \exp(\mathbf{z}_k \cdot \mathbf{z}_i)} \tag{4}$$

where $v_i, v_j \in U$ for Task 1, while $v_i \in U$ and $v_j \in T$ for Task 2, $V_B = \{v_k \in U, v_k \neq v_i\}$ for Task 1 and $V_B = T$ for Task 2. \mathbf{z}_i and \mathbf{z}_j are the embedding vectors of node v_i and node v_j respectively. For preserving the weight w_{ij} on edge (v_i, v_j) , we make the conditional distribution $p(\cdot|v_i)$ and its empirical distribution $\hat{p}(\cdot|v_i)$ coincide, and define empirical distribution as $\hat{p}(v_j|v_i) = \frac{w_{ij}}{d_i}$. Then, we minimize the following objective function:

$$O' = \sum_{v_i \in V_A} \lambda_i D(\hat{p}(\cdot|v_i)||p(\cdot|v_i)) \tag{5}$$

where $D(\cdot||\cdot)$ is the KL-divergence between two distributions, λ_i is the importance of node v_i in the network, which can be denote as the degree $d_i = \sum_i w_{ij}$, $V_A = U$ for both tasks 1 and 2. By omitting some constants, the objective function (5) can be calculated as:

$$O' = - \sum_{(v_i, v_j) \in E_v} w_{ij} \log p(v_j|v_i) \tag{6}$$

where E_v is the set of edges between V_A and V_B .

Finally, based on the objective function (6), we can complete the task 1 in the network G and task 2 in the bipartite network G_{ut} by minimizing the following objective functions:

$$O_1 = - \sum_{(u_i^x, u_j^x) \in E^x} w_{ij}^x \log p(u_j^x|u_i^x) - \sum_{(u_i^y, u_j^y) \in E^y} w_{ij}^y \log p(u_j^y|u_i^y) \tag{7}$$

$$O_2 = - \sum_{(u_i^x, t_j^x) \in E_{ut}^x} w_{ij}^x \log p(t_j^x|u_i^x) - \sum_{(u_i^y, t_j^y) \in E_{ut}^y} w_{ij}^y \log p(t_j^y|u_i^y) \tag{8}$$

4.3 Inter-network embedding

In this section, we perform the inter-network embedding on two networks G^x and G^y . Assuming that a set of anchor users bridging the two networks are available, we can learn inter-NE by the following tasks:

Task 3. (User-User Inter-NE) The target is to make the anchor and potential anchor users have coincident representations in a unified space utilizing the user transfer.

Task 4. (User-Topic Inter-NE) The target is to make the representations of the anchor and potential anchor users sharing common interests to be as similar as possible in a unified space with the assistance of topic transfer.

We first introduce the *user transfer* and *topic transfer* as follows.

User transfer will be learned across two networks G^x and G^y . To do this, a classifier (SVM) is trained for anchor link prediction [7] based on features¹ of a set of anchor users U^o , and then the results of the classifier are considered as the transfer probabilities between users. It is proved that the restrictions of probabilities are equivalent to making the representations of anchor users coincide [10]. Therefore, we define the transfer probability as $p_u(u_i^x|u_k^y)$, which represents the probability that two users u_i^x and u_k^y in different networks are the same person. Then, we get a set of seed users (anchor and potential anchor users) $U^s = \{u_k, p_u(u_k^x|u_k^y) > q\}$, where q is a transfer threshold.

Topic transfer is learned between two bipartite networks G_{ut}^x and G_{ut}^y . We follow the intuition that if many seed users who are simultaneously interested in topic t_i^x and topic t_j^y in different networks, the two topics tend to be relevant or similar [27]. Therefore, we define the topic transfer probability between topic t_i^x and topic t_j^y based on the set of seed users U^s as:

$$\begin{aligned}
 p_t(t_j^y|t_i^x) &= \sum_{u_k \in U^s} p(t_j^y|u_k)p(u_k|t_i^x) \\
 &= \sum_{u_k \in U^s} p(t_j^y|u_k) \frac{p(t_i^x|u_k)p(u_k)}{p(t_i^x)} \\
 &= \sum_{u_k \in U^s} \theta_{j,k}^y * \theta_{i,k}^x * \frac{p(u_k)}{p(t_i^x)}
 \end{aligned}
 \tag{9}$$

Where $\theta_{j,k}^y$ and $\theta_{i,k}^x$ are topic probabilities of LDA model, $p(u_k)$ is the user prior and is denoted as $p(u_k) = p_u(u_k^x|u_k^y)$, and $p(t_i^x)$ is the topic prior and is denoted as $p(t_i^x) = \sum_{u_k \in U^s} p(t_i^x|u_k)p(u_k) = \sum_{u_k \in U^s} \theta_{i,k}^x \cdot p(u_k)$.

With the assistance of user transfer, we can construct a user-user inter-network from G^x and G^y , defined as $G_{uu}^H = \{U^x \cup U^y, E_{uu}^H\}$, where E_{uu}^H is the set of social links $E^x \cup E^y$ and anchor links E_{uu}^o between anchor users. G_{uu}^H can propagate users' structural contexts across networks.

Similarly, through topic transfer we build a user-topic inter-network G_{ut}^H as the two user-topic bipartite networks G_{ut}^x and G_{ut}^y connected through the learned topic transfer in (9).

Then, we can embed two inter-networks G_{uu}^H and G_{ut}^H into a unified latent space. In particular, for the task 3, although there are no real anchor links between the potential anchor

¹The features include extended common neighbors, extended Jaccard's coefficient, extended Adamic/Adar Measure and users' topic distribution.

²Actually, the anchor links between users and topic links between topics are regarded as virtual links by user and topic transfer. The cross-network bridge nodes can be regarded as the same nodes with the help of virtual links. Therefore, the user-topic inter-network is a bipartite network, because there are only real edges between source and target nodes like user-topic intra-network.

user pairs, the information of G^x and G^y can interact with each other by the user transfer probabilities p_u^t . Therefore, we define the empirical probabilities based on p_u^t as:

$$\begin{aligned} \hat{p}(u_j^y|u_i^x) &= \sum_{u_k \in U^x} \hat{p}(u_k|u_i^x) \cdot p_u(u_j^y|u_k) \\ &= \sum_{u_k \in U^x} \frac{w_{ik}^x}{d_i^x} * p_u(u_j^y|u_k) \end{aligned} \tag{10}$$

We minimize the KL-divergence of $p(u_j^y|u_i^x)$ and $\hat{p}(u_j^y|u_i^x)$, and get the corresponding objective function:

$$O'_3 = - \sum_{(u_i^x, u_k) \in E^x} \sum_{u_j^y \in U^y} w_{ik}^x p_u^t(u_i^x|u_k) \log p(u_j^y|u_i^x) \tag{11}$$

For the task 4, although there are not real links between user u_i^x in G^x and topic t_j^y in G^y , through the topic transfer $p_t(t_j^y|t_k^x)$, user u_i^x and topic t_j^y can exchange information across networks. Therefore, we define the empirical probabilities and get the corresponding objective function as follows:

$$\begin{aligned} \hat{p}(t_j^y|u_i^x) &= \sum_{(u_i^x, t_k^x) \in E_{ut}^x} \hat{p}(t_k^x|u_i^x) p_t(t_j^y|t_k^x) \\ &= \sum_{(u_i^x, t_k^x) \in E_{ut}^x} \frac{w_{ik}^x}{d_i^x} * p_t(t_j^y|t_k^x) \end{aligned} \tag{12}$$

$$O'_4 = - \sum_{(u_i^x, t_k^x) \in E_{ut}^x} \sum_{t_j^y \in T^y} w_{ik}^x p_t(t_j^y|t_k^x) \log p(t_j^y|u_i^x) \tag{13}$$

Furthermore, with (9–13), we can calculate inter-NE by swapping the superscripts x and y , when G^y is the source network and G^x is the target network.

Finally, the task 3 and task 4 can be realized based on the objective function (11) and (13) by minimize the following two objective functions:

$$\begin{aligned} O_3 &= - \sum_{(u_i^x, u_k) \in E^x} \sum_{u_j^y \in U^y} w_{ik}^x p_u(u_i^x|u_k) \log p(u_j^y|u_i^x) \\ &\quad - \sum_{(u_i^y, u_k) \in E^y} \sum_{u_j^x \in U^x} w_{ik}^y p_u(u_i^y|u_k) \log p(u_j^x|u_i^y) \end{aligned} \tag{14}$$

$$\begin{aligned} O_4 &= - \sum_{(u_i^x, t_k^x) \in E_{ut}^x} \sum_{t_j^y \in T^y} w_{ik}^x p_t(t_j^y|t_k^x) \log p(t_j^y|u_i^x) \\ &\quad - \sum_{(u_i^y, t_k^y) \in E_{ut}^y} \sum_{t_j^x \in T^x} w_{ik}^y p_t(t_j^x|t_k^y) \log p(t_j^x|u_i^y) \end{aligned} \tag{15}$$

4.4 Joint embedding learning

The linked heterogeneous network is composed of four parts: user-user/user-topic intra-network and user-user/user-topic inter-network, where the users are shared across the four parts. To learn the representations of the networks, an intuitive idea is to collectively embed the four parts, which can be achieved by minimizing the following objective function:

$$O = O_1 + O_2 + O_3 + O_4 \tag{16}$$

We use the asynchronous stochastic gradient algorithm [19] to optimize objective (16). Optimizing objective (16) is computationally expensive, which needs to sum over the entire set of nodes, as calculating the conditional probability $p(\cdot|u_i)$. To address this problem, we adopt the negative sampling approach [14]. Take the edges whose the source node is u_i^x as a example, the equivalent counterparts can be derived, given as:

$$\log p(u_j^x|u_i^x) \propto \log \sigma(\mathbf{y}_j^{xT} \cdot \mathbf{y}_i^x) + \sum_{m=1}^K E_{u_n \sim p_n(u)} \log \sigma(-\mathbf{y}_n^{xT} \cdot \mathbf{y}_i^x) \tag{17}$$

$$\log p(t_j^x|u_i^x) \propto \log \sigma(\boldsymbol{\varphi}_j^{xT} \cdot \mathbf{y}_i^x) + \sum_{m=1}^K E_{u_n \sim p_n(u)} \log \sigma(-\boldsymbol{\varphi}_n^{xT} \cdot \mathbf{y}_i^x) \tag{18}$$

$$\log p(u_j^y|u_i^x) \propto \log \sigma(\mathbf{y}_j^{yT} \cdot \mathbf{y}_i^x) + \sum_{m=1}^K E_{u_n \sim p_n(u)} \log \sigma(-\mathbf{y}_n^{yT} \cdot \mathbf{y}_i^x) \tag{19}$$

$$\log p(t_j^y|u_i^x) \propto \log \sigma(\boldsymbol{\varphi}_j^{yT} \cdot \mathbf{y}_i^x) + \sum_{m=1}^K E_{u_n \sim p_n(u)} \log \sigma(-\boldsymbol{\varphi}_n^{yT} \cdot \mathbf{y}_i^x) \tag{20}$$

where $\sigma(x) = 1/(1 + \exp(-x))$ is the sigmoid function, K is the number of negative samples, and d_u is the output degree. We set $K = 5$ and $p_n(u) = d_u^{3/4}$ as in [14].

To minimize the (16), it is a straightforward solution to merge all kinds of edges in four sets E^x , E^y , E_{uu}^x , and E_{ut}^y together. However, because networks are heterogeneous, the weights of different types of edges cannot be comparable to each other. Therefore, it is more reasonable to alternatively sample from the four sets of edges [25], which is called joint training. Moreover, the objective function (16) can be divided into O_{uu} and O_{ut} due to respective sampling, where $O_{uu} = O_1 + O_3$ and $O_{ut} = O_2 + O_4$ are the objective function when sampling edges from E and E_{ut} , respectively. By learning the representations $\{(\mathbf{y}_i^x, \mathbf{y}_i^{x'})\}_{i=1 \dots |U^x|}$, $\{\boldsymbol{\varphi}_j^x\}_{j=1 \dots |T^x|}$, $\{(\mathbf{y}_i^y, \mathbf{y}_i^{y'})\}_{i=1 \dots |U^y|}$ and $\{\boldsymbol{\varphi}_j^y\}_{j=1 \dots |T^y|}$, we are able to represent different types of nodes with a d dimensional embedding \mathbf{y}_i^x , $\boldsymbol{\varphi}_j^x$, \mathbf{y}_i^y and $\boldsymbol{\varphi}_j^y$ in metric \mathbb{R}^d . \mathbf{y}_i^x and $\mathbf{y}_i^{y'}$ are the context representations of users as the neighbors [24].

To update the vector of nodes in network G^x , i.e., \mathbf{y}_i^x , we can calculate the gradient by sampling from E and E_{ut} , respectively. The gradient is computed as:

$$\begin{aligned} \frac{\partial O_{uu}}{\partial \mathbf{y}_i^x} &= w_{ij}^x * \{ [1 - \sigma(\mathbf{y}_j^{xT} \cdot \mathbf{y}_i^x)] \mathbf{y}_j^x - \sigma(\mathbf{y}_n^{xT} \cdot \mathbf{y}_i^x) \mathbf{y}_n^x \} \\ &+ \sum_{u_j \in U^y} w_{ik}^x * p_u(u_j^y | u_k^x) \{ [1 - \sigma(\mathbf{y}_j^{yT} \cdot \mathbf{y}_i^x)] \mathbf{y}_j^y \\ &- \sigma(\mathbf{y}_n^{yT} \cdot \mathbf{y}_i^x) \mathbf{y}_n^y \} \end{aligned} \tag{21}$$

$$\begin{aligned} \frac{\partial O_{ut}}{\partial \mathbf{y}_i^x} &= w_{ij}^x * \{ [1 - \sigma(\boldsymbol{\varphi}_j^{xT} \cdot \mathbf{y}_i^x)] \boldsymbol{\varphi}_j^x - \sigma(\boldsymbol{\varphi}_n^{xT} \cdot \mathbf{y}_i^x) \boldsymbol{\varphi}_n^x \} \\ &+ \sum_{t_j \in T^y} w_{ik}^x * p_t(t_j^y | u_k^x) \{ [1 - \sigma(\boldsymbol{\varphi}_j^{yT} \cdot \mathbf{y}_i^x)] \boldsymbol{\varphi}_j^y \\ &- \sigma(\boldsymbol{\varphi}_n^{yT} \cdot \mathbf{y}_i^x) \boldsymbol{\varphi}_n^y \} \end{aligned} \tag{22}$$

Similarly, we can obtain the partial derivatives w.r.t. the other vectors of the concerned nodes given as:

$$\begin{aligned} \frac{\partial O_{uu}}{\partial \mathbf{y}_j^x} &= w_{ij}^x * [1 - \sigma(\mathbf{y}_j^{xT} \cdot \mathbf{y}_i^x)] \mathbf{y}_i^x \\ &+ \sum_{u_j \in U^x} w_{ik}^y * p_u(u_j^x | u_k^y) [1 - \sigma(\mathbf{y}_j^{yT} \cdot \mathbf{y}_i^x)] \mathbf{y}_i^y \end{aligned} \tag{23}$$

$$\begin{aligned} \frac{\partial O_{ut}}{\partial \boldsymbol{\varphi}_j^x} &= w_{ij}^x * [1 - \sigma(\boldsymbol{\varphi}_j^{xT} \cdot \mathbf{y}_i^x)] \mathbf{y}_i^x \\ &+ \sum_{t_j \in T^x} w_{ik}^y * p_t(t_j^x | u_k^y) [1 - \sigma(\boldsymbol{\varphi}_j^{yT} \cdot \mathbf{y}_i^x)] \mathbf{y}_i^y \end{aligned} \tag{24}$$

$$\begin{aligned} \frac{\partial O_{uu}}{\partial \mathbf{y}_n^{xT}} &= w_{ij}^x * [-\sigma(\mathbf{y}_n^{xT} \cdot \mathbf{y}_i^x)] \mathbf{y}_i^x \\ &+ \sum_{u_j \in U^x} w_{ik}^y * p_u(u_j^x | u_k^y) * [-\sigma(\mathbf{y}_n^{yT} \cdot \mathbf{y}_i^x)] \mathbf{y}_i^y \end{aligned} \tag{25}$$

$$\begin{aligned} \frac{\partial O_{ut}}{\partial \boldsymbol{\varphi}_n^{xT}} &= w_{ij}^x * [-\sigma(\boldsymbol{\varphi}_n^{xT} \cdot \mathbf{y}_i^x)] \boldsymbol{\varphi}_i^x \\ &+ \sum_{t_j \in T^x} w_{ik}^y * p_t(t_j^x | u_k^y) * [-\sigma(\boldsymbol{\varphi}_n^{yT} \cdot \mathbf{y}_i^x)] \boldsymbol{\varphi}_i^y \end{aligned} \tag{26}$$

With reference to (21–26), the updating rules for network G^y can be obtained by swapping the superscripts x with y . They are not listed due to the page limit. The joint training algorithm is shown in Algorithm 1:

Algorithm 1 Joint Training

Input: two networks G^x and G^y , a set of anchor users U^o , number of negative samples K , number of samples S , learning rate η .

Output: Node representations $\{(\mathbf{y}_i^x, \mathbf{y}_i^{x'})\}_{i=1\dots|U^x|}$, $\{\boldsymbol{\varphi}_j^x\}_{j=1\dots|T^x|}$, $\{(\mathbf{y}_i^y, \mathbf{y}_i^{y'})\}_{i=1\dots|U^y|}$ and $\{\boldsymbol{\varphi}_j^y\}_{j=1\dots|T^y|}$.

```

1: Initialize  $\{(\mathbf{y}_i^x, \mathbf{y}_i^{x'})\}_{i=1\dots|U^x|}$ ,  $\{\boldsymbol{\varphi}_j^x\}_{j=1\dots|T^x|}$ ,  $\{(\mathbf{y}_i^y, \mathbf{y}_i^{y'})\}_{i=1\dots|U^y|}$  and  $\{\boldsymbol{\varphi}_j^y\}_{j=1\dots|T^y|}$ 
2: while  $iter \leq S$  do
3:   for  $m$  in  $\{x, y\}$  do
4:     Sample an edge  $(u_i, u_j)$  from  $E$  in  $G^m$ 
5:     Update  $\mathbf{y}_i, \mathbf{y}'_j$  in networks  $G^{x/y}$  based on (21) and (23) with  $\eta$ 
6:     for  $h = 0; h < K; h = h + 1$  do
7:       Sample a negative nodes  $u_n$ 
8:       Update  $\mathbf{y}_i$  and  $\mathbf{y}'_n$  in networks  $G^{x/y}$  based on (21) and (25) with  $\eta$ 
9:     end for
10:    Sample an edge  $(u_i, t_j)$  from  $E_{utt}$  in  $G^m$ 
11:    Update  $\mathbf{y}_i, \boldsymbol{\varphi}_j$  in networks  $G^{x/y}$  based on (22) and (24) with  $\eta$ 
12:    for  $h = 0; h < K; h = h + 1$  do
13:      Sample a negative nodes  $t_n$ 
14:      Update  $\mathbf{y}_i$  and  $\boldsymbol{\varphi}_n$  in networks  $G^{x/y}$  based on (22) and (26) with  $\eta$ 
15:    end for
16:  end for
17: end while

```

4.5 Mapping users across social networks

After learning the representations of users, we can discover user correspondence across social networks based on the cosine similarity, calculated using user embeddings, as follows.

$$rel(u_i^x, u_j^y) = \frac{\sum_{p=1}^d \gamma_{ip}^x \times \gamma_{jp}^x}{\sqrt{\sum_{p=1}^d \gamma_{ip}^{x^2}} \times \sqrt{\sum_{p=1}^d \gamma_{jp}^{x^2}}} \tag{27}$$

Given two sets of test users $U^x = \{u_1^x, u_2^x, \dots, u_n^x\}$ and $U^y = \{u_1^y, u_2^y, \dots, u_n^y\}$ from two social networks G^x and G^y , we compute the cosine similarity for each pair of test users from the two lists. Then, given some similarity threshold w , we return the list of user pairs³ $R = \{ \langle u_i^x, u_j^y \rangle \mid rel(u_i^x, u_j^y) > w, u_i^x \in U^x, u_j^y \in U^y \}$ as the set of discovered corresponding users.

4.6 Overall algorithm

Our overall algorithm is presented in Algorithm 2, which contains four components. First, we model the topics of user interests in G^x and G^y respectively and filter out insignificant topics from steps 1 to 4. Then, we learn the user transfer and topic transfer based on anchor users (steps 5 and 6). The details are discussed in Section 4.3. Next, in step 7, the joint

³Note that, if it is known that the two social networks are fully aligned, then for any user u_i^x with no corresponding user u_j^y such that $rel(u_i^x, u_j^y) > w$, we simply return the user u_j^y with the maximum similarity value.

training algorithm is used to learn the user and topic node representations of two networks by collectively utilizing intra-network and inter-network embedding – the detailed process is introduced in Algorithm 1. Finally, we can return a result list of predicted matching user pairs from steps 8 to 13.

Algorithm 2 Overall Algorithm

Input: two networks G^x and G^y , a set of anchor users U^o , two set of test users U^{lx} and U^{ly} , topic threshold h , similarity threshold w . **Output:** a result list R .

```

1: for  $m$  in  $\{x, y\}$  do
2:   Model the topics of user interests from the content contributed by users in the
   network  $G^m$  based on (2) and (3)
3:   Filter out insignificant topics based on  $T_i = \{k | \theta_{i,k} > h\}$  with  $h$ 
4: end for
5: Train a classifier (SVM) based on features of  $U^o$ 
6: Calculate user transfer probabilities  $p_u$  and topic transfer probabilities  $p_t$  based on the
   classifier and (9)
7: Learn node representations of two networks by joint training algorithm (Algorithm 1)
8: for each test user  $u_i^x$  in  $U^{lx}$  do
9:   for each test user  $u_j^y$  in  $U^{ly}$  do
10:    Calculate the similarity  $rel(u_i^x, u_j^y)$  based on (27)
11:    Add  $\langle u_i^x, u_j^y \rangle$  into  $R$ , if  $rel(u_i^x, u_j^y) > w$ 
12:   end for
13: end for

```

5 Experiments

In this section, we compare our LHNE with the state-of-the-art methods on two types of cross-network datasets. The first dataset is composed of a Twitter network and a Flickr network. The second dataset is a synthetic dataset including two co-author networks in Data Mining area and Wide World Web area.

5.1 Comparative methods

In this subsection, to evaluate the performance of LHNE for user identity linkage, we choose the following state-of-the-art methods as competitors, including

1. LHNE: the model proposed in this paper. LHNE is based on heterogeneous networks in terms of network structures and content. It contains User-User/User-Topic intra-NE and User-User/User-Topic inter-NE.
2. LHNE-U: a variation of our model that ignores the User-Topic inter-NE.
3. LHNE-S: a variation of LHNE based on network structures. It contains User-User intra-NE and User-User inter-NE.
4. LHNE-C: a variation of LHNE based on content. It contains User-Topic intra-NE and User-Topic inter-NE.
5. IONES: a homogeneous network embedding model for user identity linkage with “soft” constraint [10], which can simultaneously learn the follower-ship/followee-ship of each user. IONES considers only the network structures.

6. IONES-C: a variation of IONES that was extended with content information through a simple and effective manner. That is, we concatenate the user topic distribution vector and the embedding vector of a user into a long vector as the user representation for user identity linkage. IONES-C considers structural information and content information separately.
7. KNN: a popular competitive method based on k nearest neighbors search [17, 22]. In the experiments, we jointly utilize topic distribution and common neighbors as user features to compute the k nearest neighbors.

5.2 Evaluation metrics

To perform the user identity linkage, we utilize the recall, precision and F1 [5] as the metrics to evaluate the methods' performances. The recall is the fraction of the number of real corresponding user pairs that have been found over the total amount of real anchor user pairs, while the precision is the fraction of real corresponding user pairs among the result lists.

$$Recall = \frac{|Corr Pairs|}{|Real Anchor User Pairs|} \quad (28)$$

$$Precision = \frac{|Corr Pairs|}{|Result Pairs|} \quad (29)$$

$$F1 = \frac{2 * Recall * Precision}{Recall + Precision} \quad (30)$$

Where $|Real Anchor User Pairs|$ is the number of all real anchor user pairs. $|Corr Pairs|$ is the number of real corresponding user pairs that the method can find in the result list R . $|Result Pairs|$ is the number of pairs in R .

5.3 Structural characteristic metrics

We adopt two metrics (*Interop* and sparsity level) to evaluate the reliability of LHNE under the different social network structures.

Interoperability (abbreviated as *Interop*) [22] can measure the influence of overlapping of the two networks and is defined as follows:

$$Interop(x, y) = \frac{|Correlations| * 2}{|Relations^x| + |Relations^y|} \quad (31)$$

where $Relations^{x/y}$ is the set of direct edges in $G^{x/y}$. *Correlations* is the intersection of the two sets, and $0 \leq Interop(x, y) \leq 1$. When the two network are completely overlapped (or non-overlapped), *Interop*(x, y) is equal to 1 (or 0).

Meanwhile, we develop a sample ratio of edges e_s to study the influence of different sparsity levels of networks. In order to reduce the impact of other factors, we conduct variants of datasets for experiments at different sparsity levels ($e_s = [0.1, 0.2, \dots, 0.9]$) by removing overlapped edges and non-overlapped edges of two networks simultaneously, and keep the *Interop* value constant.

Besides, we also evaluate the effectiveness of methods under different w and training ratios and the sensitivities under different number of samples and dimensions.

Table 1 Statistics of social network dataset

Networks	Users	Edges	Anchors
Twitter	7118	83391	7118
Flickr	7118	23997	

5.4 Datasets

For evaluating the effectiveness, reliability and sensitivity of LHNE, We applied methods above to two types of cross-network datasets, including both social network and synthetic datasets.

Social network dataset [26] The first dataset is composed of two real social networks: Twitter and Flickr. There are 7118 anchor users with their follower/friend relationships (i.e., structural information). We collected tweets (2361.07 per user) in Twitter via Twitter API and crawled the tags (559.80 per user) in Flickr via Flickr API as user content information. The ground truth of anchor users are provided in the dataset. The basic statistics of them are shown in Table 1.

Synthetic dataset The synthetic dataset consists of two co-author networks including the co-author relationships (i.e., structural information) and paper titles from the fields of Data Mining (DM) and Wide World Web (WWW), which is constructed from Extracted DBLP Dataset [29]. We used paper titles as content information. On average, each author has 2.07 titles in DM and 1.81 titles in WWW. Because the network is directed in this paper, the co-author relationships are regarded as two directed edges with opposite directions and equal weights. There are 5353 anchor authors in synthetic dataset, forming the ground truth. The statistics of the dataset are shown in Table 2.

Analyzing two datasets above, we find the social network dataset only contains anchor user information, while the synthetic dataset includes anchor user and their non-anchor friend information simultaneously. For making our experiments more reliable, we evaluate the effectiveness of our proposed method on two datasets and the reliability and sensitivity on the synthetic dataset, because the latter contains more comprehensive user information.

5.5 Experiment results on social network dataset

In this section, we present the performance of all methods on social network dataset. LDA model is adopted to help generate the topics of user interests. The number of topics K is set to 60 and all hyperparameters are set to $1/K$. For the purpose of achieving better embeddings, we set imbalance ratio of classifier $\frac{\#negative}{\#positive} = 1$, since the classifier achieves better performance when the training sets are more balanced [7, 32].

Table 2 Statistics of synthetic dataset

Networks	Users	Edges	Anchors
DM	30795	168558	5353
WWW	28273	147932	

Table 3 Performance w.r.t topic threshold on social network dataset

Metrics	Methods	Topic threshold			
		0.1	0.3	0.5	0.7
Recall	LHNE-U	0.545	0.596	0.548	0.495
	LHNE	0.619	0.687	0.629	0.563
Precision	LHNE-U	0.422	0.498	0.434	0.353
	LHNE	0.474	0.566	0.488	0.397
F1	LHNE-U	0.476	0.538	0.484	0.412
	LHNE	0.537	0.621	0.549	0.466

Performance w.r.t topic threshold. It is critical to find the appropriate the topics of user interests that can represent users’ real interests for solving the noise problem in social networks. Therefore, we first conduct experiments to study the impact of h , which can help filter out noise of contents. Table 3 presents the performance of our proposed LHNE and LHNE-U in terms of recall, precision and F1 with different threshold h . From the results, we observe that the performance is sensitive to h . First, the performance of the two methods improves with the increase of h and achieves the best performance when $h = 0.3$. This is because the contents contain a lot of noise and the noise can be filtered with a lower threshold h . Then, the increase of h leads to the decrease of performance, as useful information is also filtered with a higher threshold. To achieve the best performance, we set $h = 0.3$.

Performance w.r.t similarity threshold and training ratio. We show the performance under different similarity threshold and training ratio settings in Figures 2 and 3, as results are very sensitive to them. As discussed in Section 4.5, the recall declines with the increase of w , since many user pairs are filtered with a higher w . Meanwhile, the user pairs with high similarities are more likely to be real corresponding user pairs. It is well known that the real corresponding user pairs usually have larger similarities than the others [5]. Therefore, the precision increases with the increase of w . To balance the recall and precision, we set $w = 0.9$.

According to Figure 2, our proposed LHNE outperforms other competitors significantly. Specifically, there is a 47.93% relative increase (0.682 vs. 0.461 recall score) comparing to IONES-C, 64.34% relative increase (0.682 vs. 0.415 recall score) comparing to IONES and 182.99% relative increase (0.682 vs. 0.241 recall score) comparing to KNN when $w = 0.9$. By taking a closer look at the dataset, we notice that there are 40.94% users without any

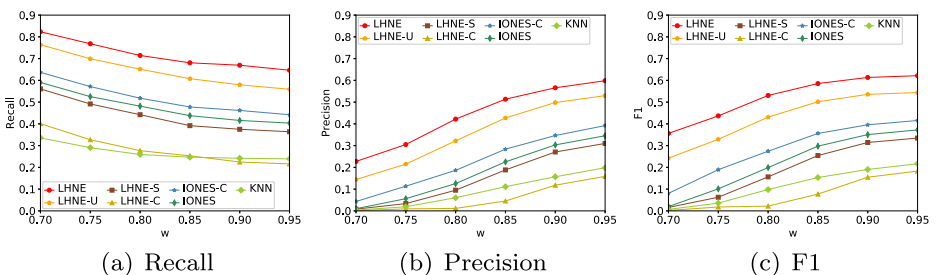


Figure 2 Performance w.r.t similarity threshold on social network dataset

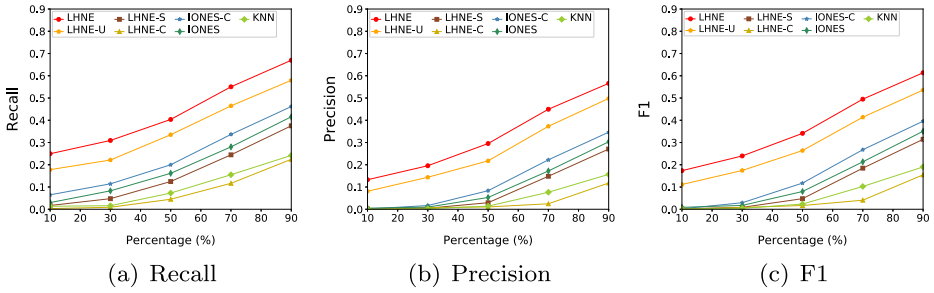


Figure 3 Performance w.r.t training ratio on social network dataset

links to other users in Flickr network and 18.45% users in Twitter network. The loss of structural information degenerates the performance of methods. LHNE can solve this problem by linking the topics and users because of the correlation of information, therefore, the missing information between users can be supplemented via topic nodes serving as the context of user nodes. In contrast, IONES-C and KNN consider content and structural information separately and IONES only considers structural information, so that they fail to correlate users without social links. Meanwhile, LHNE exploits the transfer across networks in terms of users and topics. With the help of user transfer and topic transfer, the representations of users are more comprehensive and effective. Consequently, LHNE has better performance than LHNE-U, since LHNE-U only considers user transfer across networks. Besides, it can be concluded that the structural information is more discriminative than the content information, as LHNE-S outperforms LHNE-C. In the meantime, LHNE performs better than LHNE-S (with only structural information) and LHNE-C (with only content information), showing the benefits brought by jointly leveraging structural and content information.

Additionally, Figure 3 presents LHNE outperforms other methods with different training ratios. It can be observed that LHNE achieves much higher recall, precision and F1 even when the given ratio is as low as 10% to 20%, indicating that LHNE can capture most common knowledge for user identity linkage by jointly leveraging structural and content information. It is significant for real social networks without a lot of training data. Considering the performance of the competitors, we set training ratio as 0.9.

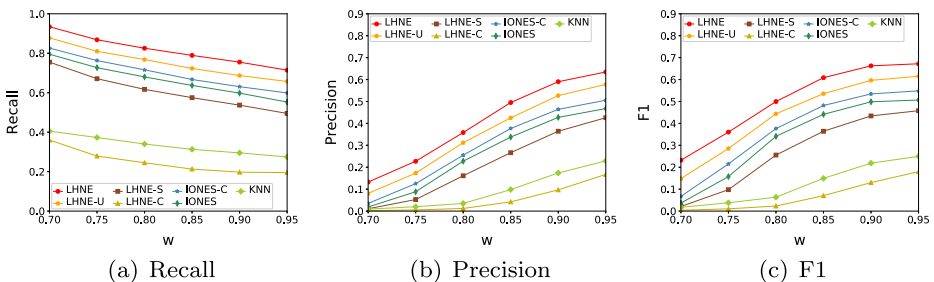


Figure 4 Performance w.r.t similarity threshold on synthetic dataset

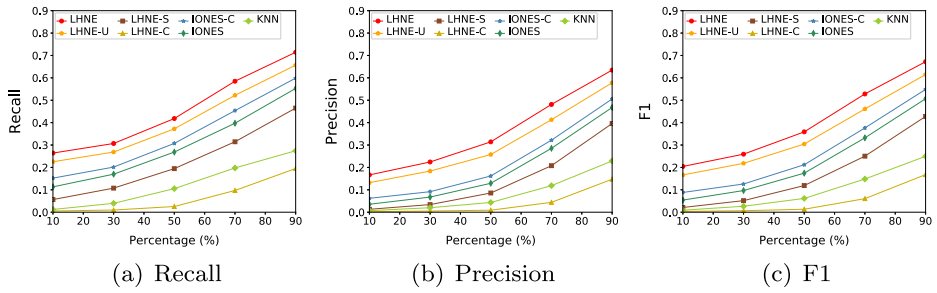


Figure 5 Performance w.r.t training ratio on synthetic dataset

5.6 Experiment results on synthetic dataset

In this experiment, we focus on the reliability of LHNE on different network structures (e.g., *Interop* and sparsity) and parameter sensitivity (e.g., the number of samples and dimension) besides effectiveness. The parameter settings of K , hyperparameters and imbalance ratio are as same as social network dataset.

Performance w.r.t similarity threshold and training ratio. Figures 4 and 5 reports the performance of all methods on synthetic dataset. We can see that the comparison result is similar to that presented in Figures 2 and 3. LHNE performs better than the other methods under different w and training ratio. However, there are two different issues between Figures 4 and 5 and Figures 2 and 3. Firstly, all methods on synthetic dataset perform better than on social network dataset. This is because synthetic dataset has a better network structure. Specifically, synthetic dataset includes anchor users and non-anchor users simultaneously and the ratio of edges of the DM and WWW networks is more balanced than those of the social network dataset, as shown in Table 1. Secondly, we also observe that LHNE outperforms IONES-C and IONES more greatly on social network dataset than on the synthetic dataset. Because of the nature of social networks, the social network datasets has more noise than the synthetic dataset. LHNE is robust by filtering noise of content with a suitable topic threshold h . Moreover, Joint Embedding of structural and content information is more stable for LHNE. Similar to the considerations on the social network dataset, we set w as 0.95 and the training ratio as 0.9.

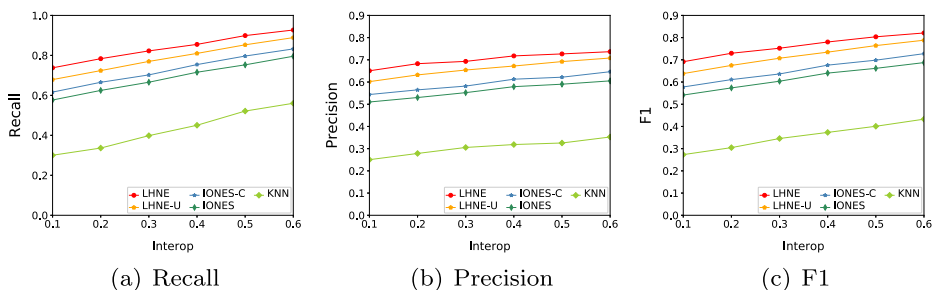


Figure 6 Reliability w.r.t *Interop* on synthetic dataset

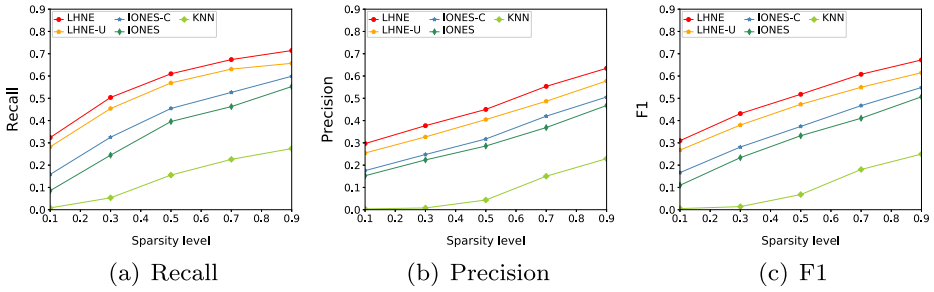


Figure 7 Reliability w.r.t sparsity on synthetic dataset

Reliability w.r.t *Interop* and sparsity. Because all methods are based on network structures, they often suffer from problems such as non-overlap and sparsity. Therefore, we explore the performance for different *Interop* and sparsity level. Figures 6 and 7 show the results. The *Interop* of the original synthetic dataset is 0.0889. In this experiment, we vary the *Interop* value from 10% to 60% by removing edges of anchor users. According to Figure 6, all methods depend on the *Interop*, and they achieve better performance as the *Interop* value increases. Moreover, LHNE outperforms all competitors, even when the *Interop* value is low. Figure 7 shows LHNE performs better than other embedding-based methods, even in the cases where the sparsity level is lower than 30%. In other words, LHNE can achieve good performance with relatively less structural information. It can be concluded that linking topics and users can make LHNE more reliable. To maintain the original network structures, we set the *Interop* as 0.0889 and the sparsity level as 1.

Sensitivity w.r.t number of samples and dimension. For embedding-based methods, the number of samples and the dimension have a significant impact on the computational speed and storage. Therefore, we analyze the converging performance by varying the number of samples and the performance by varying d . Figure 8 shows LHNE converges much faster than IONES-C and IONES. We believe the gain comes from the contribution of joint embedding of structural and content information. It is beneficial for real-time applications based on user identity linkage. Therefore, considering the real-time requirement and the convergence of the competitors, we set the number of samples as 10 million when all methods converge. Besides, Figure 9 presents that LHNE achieves much higher performance than other methods even when d is low. Therefore, more efficient computation in term of

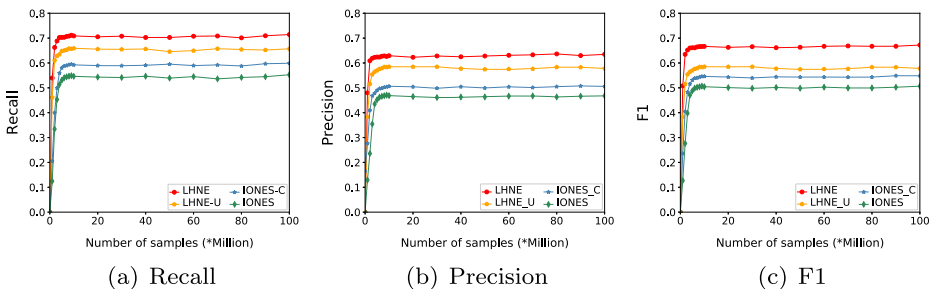


Figure 8 Sensitivity w.r.t number of samples on synthetic dataset

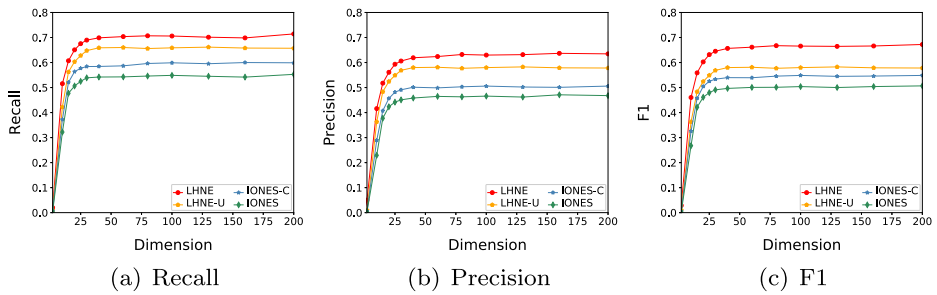


Figure 9 Sensitivity w.r.t dimension on synthetic dataset

time and space can be realized through lower dimensional embedding. We set $d = 100$ when all the methods are stable and can obtain best performance.

6 Conclusion

In this paper, we aim to learning the comprehensive representations of users considering the fact that structural and content information are correlative, and propose a linked heterogeneous network embedding method for user identity linkage to address the challenging issues, including heterogeneity of information, diversity of social networks and noise. We conducted extensive experiments to evaluate the performance of LHNE on both real social network and synthetic datasets. The results showed LHNE is significantly better than the state-of-the-art methods (up to 47.93% enhancement comparing to IONES-C), when there are 40.94% and 18.45% users without any links to others in Twitter and Flickr network, respectively. Therefore, our model can work well even with little or no structural information, when data acquisition is difficult in social networks because of privacy protection. The performance of LHNE can be reinforced by fully exploring the correlation between heterogeneous information, which can provide complementary information to each other. For future works, we may consider integrating more types of information into LHNE for improving the performance of embedding and extending the model to multiple networks.

References

1. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003)
2. Cai, H., Zheng, V.W., Chang, K.C.C.: A comprehensive survey of graph embedding: Problems, techniques and applications. arXiv:1709.07604 (2017)
3. Cao, X., Yu, Y.: Bass: a bootstrapping approach for aligning heterogeneous social networks. In: *ECML PKDD*, pp. 459–475 (2016)
4. Cao, X., Yu, Y.: Joint user modeling across aligned heterogeneous sites. In: *Recsys*, pp. 83–90 (2016)
5. Chen, W., Yin, H., Wang, W., Zhao, L., Hua, W., Zhou, X.: Exploiting spatio-temporal user behaviors for user linkage. In: *CIKM*, pp. 517–526 (2017)
6. Dong, Y., Tang, J., Wu, S., Tian, J., Chawla, N.V., Rao, J., Cao, H.: Link prediction and recommendation across heterogeneous social networks. In: *ICDM*, pp. 181–190 (2012)
7. Kong, X., Zhang, J., Yu, P.S.: Inferring anchor links across multiple heterogeneous social networks. In: *CIKM*, pp. 179–188 (2013)
8. Korula, N., Lattanzi, S.: An efficient reconciliation algorithm for social networks. pp. 377–388 (2014)
9. Liu, S., Wang, S., Zhu, F., Zhang, J., Krishnan, R.: Hydra: large-scale social identity linkage via heterogeneous behavior modeling. In: *ACM SIGMOD*, pp. 51–62 (2014)

10. Liu, L., Cheung, W.K., Li, X., Liao, L.: Aligning users across social networks using network embedding. In: IJCAI, pp. 1774–1780 (2016)
11. Malhotra, A., Totti, L., Meira, J.R. W., Kumaraguru, P., Almeida, V.: Studying user footprints in different online social networks. In: ASONAM, pp. 1065–1070 (2012)
12. Man, T., Shen, H., Huang, J., Cheng, X.: Context-adaptive matrix factorization for multi-context recommendation. In: CIKM, pp. 901–910 (2015)
13. Man, T., Shen, H., Liu, S., Jin, X., Cheng, X.: Predict anchor links across social networks via an embedding approach. In: IJCAI, vol. 16, pp. 1823–1829 (2016)
14. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
15. Mu, X., Zhu, F., Lim, E.P., Xiao, J., Wang, J., Zhou, Z.H.: User identity linkage by latent user space modelling. In: KDD, pp. 1775–1784 (2016)
16. Narayanan, A., Shmatikov, V.: De-anonymizing social networks. In: ISSP, pp. 173–187 (2009)
17. Nie, Y., Jia, Y., Li, S., Zhu, X., Li, A., Zhou, B.: Identifying users across social networks based on dynamic core interests. *Neurocomputing* **210**, 107–115 (2016)
18. Phan, M.C., Sun, A., Tay, Y.: Cross-device user linking: url, session, visiting time, and device-log embedding. In: SIGIR, pp. 933–936 (2017)
19. Recht, B., Re, C., Wright, S., Niu, F.: Hogwild: a lock-free approach to parallelizing stochastic gradient descent. In: NIPS, pp. 693–701 (2011)
20. Riederer, C., Kim, Y., Chaintreau, A., Korula, N., Lattanzi, S.: Linking users across domains with location data: theory and validation. In: WWW, pp. 707–719 (2016)
21. Shu, K., Wang, S., Tang, J., Zafarani, R., Liu, H.: User identity linkage across online social networks: a review. *ACM SIGKDD Explorations Newsletter* **18**(2), 5–17 (2017)
22. Tan, S., Guan, Z., Cai, D., Qin, X., Bu, J., Chen, C.: Mapping users across networks by manifold alignment on hypergraph. In: AAAI, vol. 14, pp. 159–165 (2014)
23. Tang, J., Qu, M., Mei, Q.: Pte: predictive text embedding through large-scale heterogeneous text networks. In: KDD, pp. 1165–1174 (2015)
24. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: large-scale information network embedding. In: WWW, pp. 1067–1077 (2015)
25. Xie, M., Yin, H., Wang, H., Xu, F., Chen, W., Wang, S.: Learning graph-based Poi embedding for location-based recommendation. In: CIKM, pp. 15–24 (2016)
26. Yan, M., Sang, J., Mei, T., Xu, C.: Friend transfer: cold-start friend recommendation with cross-platform transfer learning of social knowledge. In: ICME, pp. 1–6 (2013)
27. Yan, M., Sang, J., Xu, C., Hossain, M.S.: Youtube video promotion by cross-network association: @britney to advertise gangnam style. *TMM* **17**(8), 1248–1261 (2015)
28. Yan, M., Sang, J., Xu, C., Hossain, M.S.: A unified video recommendation by cross-network user modeling. *TOMM* **12**(4), 53 (2016)
29. Yang, D., Xiao, Y., Tong, H., Cui, W., Wang, W.: Towards topic following in heterogeneous information networks. In: ASONAM, pp. 363–366 (2015)
30. Zafarani, R., Liu, H.: Connecting corresponding identities across communities. pp 354–357 (2009)
31. Zhang, J., Philip, S.Y.: Integrated anchor and social link predictions across social networks. In: IJCAI, pp. 2125–2132 (2015)
32. Zhang, J., Kong, X., Philip, S.Y.: Predicting social links for new users across aligned heterogeneous social networks. In: ICDM, pp. 1289–1294 (2013)
33. Zhang, J., Kong, X., Yu, P.S.: Transferring heterogeneous links across location-based social networks. In: WSDM, pp. 303–312 (2014)
34. Zhang, J., Yu, P.S., Zhou, Z.H.: Meta-path based multi-network collective link prediction. In: KDD, pp. 1286–1295 (2014)