

Article

JARUA: Joint Embedding of Attributes and Relations for User Alignment across Social Networks

Min Yang, Baiyang Chen  and Xiaoliang Chen *

School of Computer and Software Engineering, Xihua University, Chengdu 610039, China

* Correspondence: chexiaol@iro.umontreal.ca

Abstract: User alignment (UA), a central issue for social network analysis, aims to recognize the same natural persons across different social networks. Existing studies mainly focus on the positive effects of incorporating user attributes and network structure on UA. However, there have been few in-depth studies into the existing challenges for the joint integration of different types of text attributes, the imbalance between user attributes and network structure, and the utilization of massive unidentified users. To this end, this paper presents a high-accuracy embedding model named Joint embedding of Attributes and Relations for User Alignment (JARUA), to tackle the UA problem. First, a mechanism that can automatically identify the granularity of user attributes is introduced for handling multi-type user attributes. Second, a graph attention network is employed to extract the structural features and is integrated with user attributes features. Finally, an iterative training algorithm with quality filters is introduced to bootstrap the model performances. We evaluate JARUA on two real-world data sets. Experimental results demonstrate the superiority of the proposed method over several state-of-the-art approaches.

Keywords: user alignment; social networks; multi-granularity; iterative training



Citation: Yang, M.; Chen, B.; Chen, X.

JARUA: Joint Embedding of Attributes and Relations for User Alignment across Social Networks.

Appl. Sci. **2022**, *12*, 12709. <https://doi.org/10.3390/app122412709>

Academic Editors: Carson K. Leung, Fei Hao, Giancarlo Fortino and Xiaokang Zhou

Received: 4 November 2022

Accepted: 9 December 2022

Published: 11 December 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Social networks play an essential role in our daily life. People join multiple social network platforms, e.g., Facebook and Instagram, to enjoy different services simultaneously. Platform users usually maintain various accounts in different social networks, acting as the bridges connected between multiple networks. The user alignment (UA) problem, which aims to recognize the same natural person across different social networks, has attracted substantial research attention due to its significant research challenges and practical value. For example, UA across multiple social networks can help comprehensively understand users' characteristics, thus helping Internet service providers offer better services. It also provides the potential for solving problems that cannot be completed within a single network, such as cold start and data sparsity in many prediction tasks. In addition, UA allows us to analyze user migration patterns and guide web application development [1]. Therefore, UA benefits significantly in many social network applications, including item recommendation [2], link prediction [3], information diffusing prediction [4], and network dynamics analysis [5].

Intuitively, the user's unique attributes, e.g., email addresses and identity numbers, are adequate for aligning the same users among multiple social networks if privacy issues are not considered. As these privacy-sensitive attributes are usually not authorized to be exposed online, many studies have focused on leveraging other accessible user attributes that appear on social platform pages. Multi-modality user attributes have been exploited for UA tasks, such as text [6,7], image [8,9], geographic location [10,11], etc. The most widely used and accessible attribute modality is text. Researchers either adopt heuristic rules to define similarity measures of text attributes or build representation learning (RL) models to compare distances of user features. The latter is currently the mainstream. A

number of RL techniques including autoencoders (AE) [7], convolution neural networks (CNN) [6] and recurrent neural networks (RNN) [12] have been adopted for extracting text features. However, such methods generally take a single method to encode the attribute text (e.g., one-hot encoding, bag-of-word vectors, and TF-IDF), which ignores the characteristics of different types of text. A perfect example can be observed in the following four types of attributes. Usernames often have a large proportion of customized words with special symbols, e.g., “S.Joshi” and “♣Joshi♣”. Age is generally represented by an integer, e.g., 18 and 60. Affiliation is usually a phrase that consists of some regular words, e.g., the full name of a corporation or a university. Blog/post may include many paragraphs rich with spelling mistakes, slang, or domain-specific jargon. **(Challenge 1) There still exists a challenge in jointly integrating different types of text attributes.** Generally, text data, regardless of its length, contain a coarse- to fine-grained hierarchical structure. For example, letters can be combined together to form words, and phrases can be decomposed into words. In other words, granular structure naturally exists in texts. If we can find a proper representation method that conforms to the inner structure of the text attribute, it would be more efficient to capture the text features. This work attempts to handle multi-type text attributes in a multi-granularity approach which automatically discovers the inner granularity structure of attribute texts.

Another line of research for UA utilizes the structural information of social networks. More current methods incorporate both user attributes and network structures to enhance the UA performance [13,14]. For instance, the model REGAL [13] captures the node’s structural identity and attribute identity by a matrix factorization-based approach. Another typically related study on UA feature combination is undertaken in CoLink [15], which iteratively learns an attribute-based and a structure-based model with a co-training strategy. However, **(Challenge 2) it is not a trivial task to effectively leverage user attributes and network structures to address UA problems when they are quite imbalanced.** The attributes and relationships of users in real social networks can be diverse. Some users offer little profiles on their home pages but have rich connections with others. On the contrary, a number of users on LinkedIn give detailed personal information but they do not like to interact with others. This imbalance results in the low accuracy of user identification algorithms. If either of the two feature types is too faulty, it may lead to error propagation or additional noise. An example is shown in Figure 1. If the actually existing relations r_6 and r'_2 are missing, user u_2 can be easily aligned to u'_1 by mistake. Similarly, the linkage error might be propagated to the other pair u_1 and u'_2 with the help of relations r_1 and r'_1 . Moreover, it is also possible for a careless algorithm to determine incorrectly that u_5 and u'_5 are not the same natural person by comparing their names “Joshi 001” and “S. Joshi”, or their affiliations “AA Corp.” and “University of AAA”. Therefore, attributes and relations have different levels of importance in UA tasks when they are considered imbalanced. Distinguishing the contributions of attribute information and structural information benefits a great deal from enhancing the linking accuracy of UA models. This paper adopts an embedding-based approach to incorporate user attributes and network structures into a unified optimization framework, which ensures the robustness of the model.

Supervised UA approaches that build learning models on a set of identified user pairs have brought convenience to train highly accurate models. The same technologies have also created some potential problems that are exposed in almost every supervised work. The alignment accuracy is limited by the quantity of pre-aligned user pairs (labeled data) across networks, while massive unidentified users (unlabeled data) are not utilized. In the meantime, pre-aligned user pairs are often insufficient due to costly manual annotations and the huge scale of social networks. **(Challenge 3) Making the most of unlabeled data for the alignment task remains a challenge that has not yet been fully resolved.** A straightforward solution for exploiting unlabeled data is to predict new auto-labeled data iteratively based on initial training samples [15,16]. To improve the quality of auto-labeled data, this work introduces quality filters based on the symmetry property of similarity function and adopts a threshold function to obtain higher alignment accuracy.

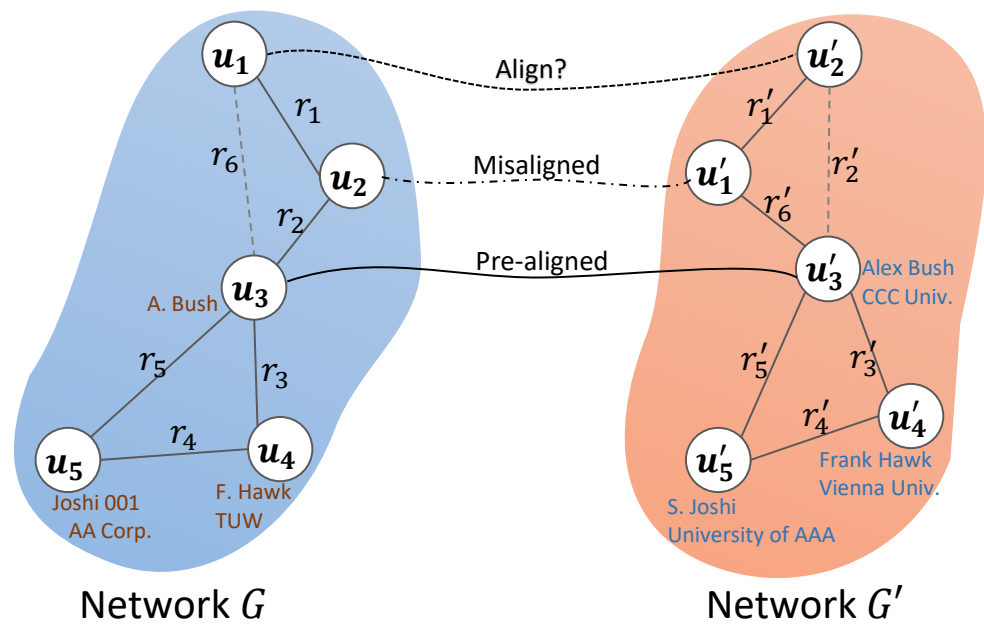


Figure 1. An example of user alignment across social networks. G and G' denote two networks to be aligned. The lines between users within each network represent the relations between users, where solid lines (r_1 and r'_1 , etc.) and dashed lines (r_6 and r'_2) represent actual existing user relations and potential but not observed relations, respectively. The lines between two users across G and G' denote the UA problem. The solutions determine whether potential user relations actually exist or not.

To address the aforementioned three challenges in the UA problem, this paper proposes a novel embedding-based UA model, Joint embedding of Atttributes and Relations for User Alignment (JARUA). The goal of JARUA is to map two social networks into a common latent space where aligned users are close to each other. In particular, all users' text attributes are first automatically divided into the subword level and the word level to unify the modeling of multi-type text attributes. Second, JARUA employs a graph attention network (GAT) [17] integrated with multi-granular attribute embedding to effectively leverage the attributes and network structure, based on which a unified objective function is designed for optimization. Third, an iterative training algorithm is developed to improve the model performance by making full use of unlabeled data. The algorithm trains the model on an initial set of labeled data and iteratively predicts new auto-labeled data, which are added to the training set for the next iteration. After each iteration, filtering measures are taken to ensure the quality of auto-labeled samples. The main contributions of this work include:

- A novel embedding-based UA model to incorporate attributes and network structure into a unified optimization framework which is robust to user attributes missing or sparse networks.
- A multi-granularity attribute representation method is developed to effectively process multi-type user attributes.
- An iterative training algorithm with filtering mechanics, which utilizes numerous unlabeled data to bootstrap the model performances.
- Extensive experiments on two real-world datasets demonstrate the superiority of JARUA over a series of comparison methods.

2. Related Work

UA was initially formalized as connecting corresponding user identities across different communities in [18], which is also called user identity linkage (UIL), user identification, anchor link prediction, etc. Designing UA models aims to align the social accounts belonging to identical natural person among multiple social media platforms. These pairs

of identical users who act as anchors connecting various networks are known as aligned user pairs.

This section attempts to provide a brief summary of the literature relating to UA issues. Existing UA methods can be roughly classified into three categories: supervised, semi-supervised, and unsupervised approaches. Most research is supervised, where the main idea is training a ranking model or a binary classifier to find potential identical users by using pre-matched user pairs as guidance [3,6,19–29]. For example, Man et al. [22] proposed an anchor link prediction method PALE for preserving significant structural regularities of networks with an awareness of supervised pre-aligned users. Likewise, Mu et al. [23] provided an approach ULink, which attempts to map the users from multiple social networks into the same latent user space through a projection matrix. Zhang et al. [6] employed a graph neural network (GNN) to directly incorporate target users' ego networks into an embedding space by presenting a model MEgo2vec. Liu et al. [25] represented the social structure by determining the weighted contribution probabilities of friendships based on an attention-based network embedding (ABNE). Qiao et al. [26] introduced the UA model SAUIL, a Siamese neural network, to learn the high-level representation of users' web-browsing behaviors. Li et al. [27] extracted the structural information redundancies by analyzing the similarities of k -hop neighbors and integrated them with name-based attribute features for user identification. Fu et al. [28] exploited the higher-order structural properties and alignment-oriented structural consistency to learn a unified graph embedding method (MGGE), which aimed to learn feature vectors of the graph. A recent study by Li et al. [29] considered UA as a sequential decision problem and proposed a reinforcement learning model to align users from a global perspective.

The above published supervised studies on UA issues depend heavily on the quantity and quality of labeled data. However, collecting sufficient aligned user pairs as annotations is inevitable. Some unsupervised methods are capable of addressing the UA problem without any labeled data [13,15,30–36]. Liu et al. [30] proposed an improved n -gram model that can automatically generate training data according to an evaluation of the username rareness. Then, a binary classifier was trained to recognize identical users. Lacoste-Julien et al. [32] highlighted the function of heuristic string similarities to align user attributes through simple greedy matching (SiGMA). Some researchers have focused on feature mapping or distance optimization. Nie et al. [33] suggested an approach of dynamic core interests (DCIM) that integrates network structure and user-generated contents to align users. Li et al. [31] recommended the model UUIL that can utilize earth mover's distance to measure the distance between user degree distributions. A user identification algorithm FRUI-P was reported by Zhou et al. [35], which introduced the concept of "friend relationships" without prior knowledge for solving the UA problem. Zhou et al. [36] proposed capturing node distribution in Wasserstein space and reformulating the UA task as an optimal network transport problem in a fully unsupervised manner. Recently, Li et al. [37] studied user's check-in records, and jointly considered user's spatial-temporal information (e.g., location and time) to link identical users without any annotations.

Unsupervised approaches usually suffer from lower performances than supervised ones, although unsupervised methods do not rely on labeled data. Semi-supervised operations offer an effective way of leveraging unlabeled samples along with a few annotations. Therefore, many semi-supervised approaches [16,38–45] have recently emerged to tackle the UA problem. The typical feature of semi-supervised UA models is that newly aligned user pairs can be predicted during the training process. The energy-based model COSNET, proposed by Zhang et al. [16], detailed that joint integration of the local user matching, network structure, and global consistency is beneficial to UA models. Similarly, Liu et al. [38,44] embedded user relations into input/output context vectors by employing a few annotations. In a follow-up study, the approach FRUI established by Zhou et al. [40] iteratively linked user identities by calculating a structure-based matching degree. Li et al. [42] designed the weakly supervised adversarial learning model SNNA, to recognize users at the distribution level. Later, the extension model MSUIL [43] based on SNNA was presented to address

the challenges in multiple social network scenarios. Liu et al. [45] recently studied the mutual promotion effect of users and employers in heterogeneous social networks, and proposed the matrix factorization-based representation learning framework MFRep for network alignment. In all the semi-supervised studies reviewed here, the proper combination of attributes and structure information is recognized as a potential solution to the UA problem.

3. Problem Formulation

This section formalizes the problem addressed in this paper. Basically, a social network can be represented in a three-tuple format $G = (U, R, A)$, where $U = \{u_1, u_2, \dots, u_n\}$ is the set of n users; $R = \{(u_i, u_j) | u_i, u_j \in U\}$ represents the set of relations among users, e.g., friendships on Facebook; and $A = \{a_{u_1}, a_{u_2}, \dots, a_{u_n}\}$ is the set of users' text attributes, e.g., name, affiliation, and education experiences. Each text attribute of a user is represented as a key–value pair, denoted by $p = \langle \text{key} : \text{value} \rangle$. The relationships among users are considered undirected in this paper.

Definition 1. Multi-type attributes: Let $G = (U, R, A)$ be a social network with a set of text attributes A . Each user has multiple attributes, denoted by $a_{u_i} = \{p_1, p_2, \dots, p_m\}$. If $\exists p_j, p_k \in a_{u_i}$, $p_j = \langle \text{key}_j : \text{value}_j \rangle$, $p_k = \langle \text{key}_k : \text{value}_k \rangle$, and value_j and value_k are different in terms of text composition, then a_{u_i} is called the multi-type attributes for the user u_i .

A concrete example of multi-type attributes is as follows. A user registers their name “Michael” on a social network. Then, he has the attribute $p = \langle \text{name:Michael} \rangle$, where “name” is the key (type) of the attribute, and “Michael” is the attribute value. The user may also have other attributes of different types, such as nickname, age, affiliations, etc. We suppose that Michael filled out the following information on a social media platform: $p_1 = \langle \text{name:Michael} \rangle$, $p_2 = \langle \text{nickname:Colorful-peak} \rangle$, $p_3 = \langle \text{age:28} \rangle$, and $p_4 = \langle \text{affiliation:Byerly Hall, 8 Garden Street, Cambridge} \rangle$. That is to say, he has the multi-type attributes $a_{u_i} = \{p_1, p_2, p_3, p_4\}$.

Definition 2. Word granularity factor: Let \mathcal{V} be a vocabulary set composed of all words in a certain language. Given a piece of text T consisting of basic tokens (e.g., words, letters, numbers, and other symbols) in that language, the word granularity factor $\omega(T)$ is the proportion of the count number of word tokens to that of all basic tokens in T , defined as:

$$\omega(T) = \frac{|\mathcal{V}| \cap |f(T)|}{|f(T)|}, \quad (1)$$

where $f(\cdot)$ is a tokenization function that splits texts into basic tokens.

Table 1 gives some examples of calculating the word granularity factors. We take the first sample to illustrate the calculation. $T = “\clubsuit\text{Joshi}\clubsuit”$ is a piece of text that composed of a word “Joshi” and two same symbols “ \clubsuit ”. Hence, we have the basic tokens $f(T) = \{\clubsuit, \text{Joshi}, \clubsuit\}$ and the word token Joshi. The word granularity factor can be obtained according to $\omega(T) = (1/3) = 0.33$.

Table 1. Examples of the word granularity factor

Sample Texts	Basic Tokens	ω
$\clubsuit\text{Joshi}\clubsuit$	$\clubsuit, \text{Joshi}, \clubsuit$	0.33
Joshi0001	Joshi, 0001	0.50
University of Florida	University, of, Florida	1.00
A hierarchical three way interconnect architecture for hexagonal processors	A, hierarchical, three, way, interconnect, architecture, for, hexagonal, processors	1.00

This paper considers such a scenario: some users are involved in two different social networks which form the anchor links across the networks. The source network is denoted as $G^X = (U^X, R^X, A^X)$, the target as $G^Y = (U^Y, R^Y, A^Y)$, and vice versa. This investigation intends to identify, if any, the counterpart in G^Y (or G^X) for each user in G^X (or G^Y). The UA problem can be formalized as follows.

Definition 3. User alignment: given two arbitrary social networks G^X and G^Y and a few pre-aligned user pairs $P = \{(u_i, u_j) | u_i \in U^X, u_j \in U^Y\}$, user alignment (UA) aims to find the other hidden aligned user pairs $Q = \{(u_i, u_j) | u_i \in U^X, u_j \in U^Y, (u_i, u_j) \notin P\}$, where u_i and u_j belong to a same natural person.

4. Methodology

4.1. Overview

This section details the method of Joint embedding of Attributes and Relations for User Alignment (JARUA), which includes three components: multi-granularity attribute embedding, attention-based structure embedding, and iterative training process. First, all text attributes of users are automatically divided into two granularities: the subword level and the word level. Simple and intuitive multi-layer perceptrons (MLPs) are then designed to extract attribute features of each granularity level. Second, a GAT approach is employed to capture users' structural characteristics, and the attribute and structure features are combined to achieve the embeddings of users, based on which a unified objective function is designed for optimization. Third, a large number of unlabeled samples is utilized because of the expected difficulty in obtaining adequate labeled data on the UA problem. We outline an iterative training algorithm to bootstrap the model performances using the unlabeled samples. Finally, each source user's alignment can be found by searching the nearest neighbors in the target network. Figure 2 depicts the overall framework of our proposed approach. The following parts of this paper focus on the elaboration of the multi-granularity attribute embedding, the attention-based structure embedding, and the iterative training algorithm.

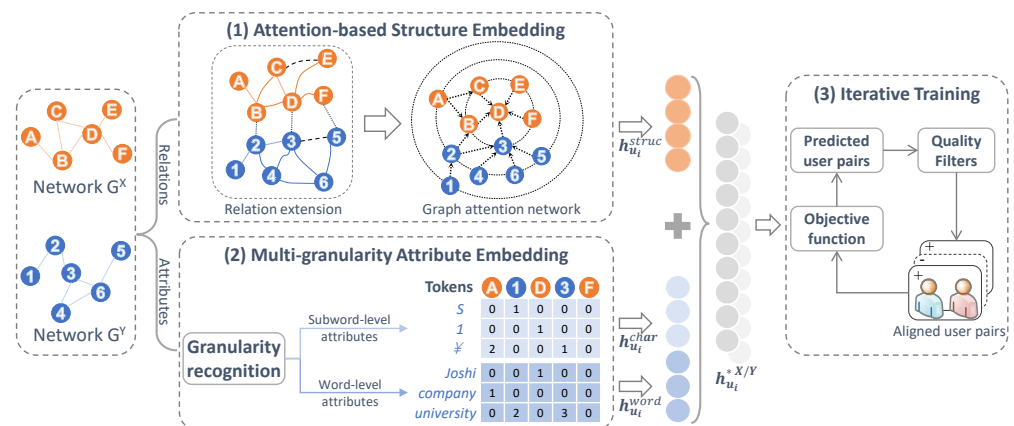


Figure 2. The overview of JARUA framework.

4.2. Multi-Granularity Attribute Embedding

Granular computing (GrC) is an emerging information processing paradigm for solving complex problems. The concept of granularity is known as the degree of abstraction or detail. In NLP, the text granularity can be divided into many levels, such as character level, word level, sentence level, paragraph level, etc. The great majority of UA literature involving text is based on one single text granularity. Besides, the length of attribute texts has a large span. The shorter texts have only a few characters (e.g., <nickname:tJoet >), while the longer texts may comprise several pages. However, configuring more massive granularity levels may fail to process shorter texts. A classic example of the excessive granularity size selection is that the sentence level is difficult to work on with nicknames.

This paper only considers the subword-level and word-level granularities. The rest of the section begins with an analysis of how to recognize the granularity level of an attribute type. Then, we introduce the embedding methods corresponding to each granularity level.

4.2.1. Attribute Granularity Recognition

In order to determine whether a text belongs to the subword level or the word level, we define an indicator function $\mathbb{I}(\cdot)$ as:

$$\mathbb{I}(x) = \begin{cases} 1, & \text{if } \omega(x) > \beta, \\ 0, & \text{otherwise.} \end{cases} \quad (2)$$

In the above function, $\beta \in [0, 1]$ is a parameter denoting the preference for a certain granularity level of the texts. In this paper, we empirically set $\beta = 0.8$. This function means that if some text's word granularity factor is bigger than β , the text is treated at the word-level granularity. We can observe that the text "University of Florida" shown in Table 1 belongs to the word-level granularity, as its word granularity factor equals 1.

For a social network with multi-type attributes, we aim to recognize the granularity level to which an attribute type belongs. Formally, for each attribute type k , we first perform random sampling and obtain a set $U' = \{u_1, u_2, \dots, u_S\} \subset U$ of S users. Their attribute values are combined to form the set T' of a piece of new text as:

$$T' = g(k) = \{value_j | key_j = k, p_j \in a_{u_i}, u_i \in U'\} \quad (3)$$

Then, we define another indicator function $\mathbb{I}'(\cdot)$ to determine the granularity level of the attribute type k .

$$\mathbb{I}'(k) = \begin{cases} 1, & \text{if } \mathbb{I}(g(k)) = 1, \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

If the values of an attribute type k are word-level granular, then the function $\mathbb{I}'(\cdot)$ will recognize the type as the word-level attribute.

The above definitions support the description of user u_i 's subword-level and word-level attributes, which can be represented as $a_{u_i}^c = \{p_j = \langle key_j : value_j \rangle | \mathbb{I}'(key_j) = 0\}$ and $a_{u_i}^w = \{p_j = \langle key_j : value_j \rangle | \mathbb{I}'(key_j) = 1\}$, respectively. Correspondingly, the subword-level and word-level attribute values of user u_i can be described by $v_{u_i}^c = \{value_j | p_j \in a_{u_i}^c, \mathbb{I}'(key_j) = 1\}$ and $v_{u_i}^w = \{value_j | p_j \in a_{u_i}^w, \mathbb{I}'(key_j) = 0\}$, respectively.

4.2.2. Subword-Level Attribute Embedding

This part examines how to extract subword-level attribute features. First, each user's subword-level attribute value $v_{u_i}^c$ is tokenized into a set of basic tokens, including letters, numbers, special symbols (e.g., "♣" and "+") and q -grams [46] (i.e., several characters that frequently occur together, e.g., "th" and "ing"). Then, we count the total number of each basic token. As a result, the subword-level attribute value of a user u_i can be represented by a bag-of-words vector $\vec{v}_{u_i}^c \in \mathbb{R}^{1 \times \mathcal{N}}$ as:

$$\vec{v}_{u_i}^c = [count_1^i, count_2^i, \dots, count_t^i, \dots, count_{\mathcal{N}}^i] \quad (5)$$

where $count_t^i$ is the count number of the token t in $v_{u_i}^c$, and \mathcal{N} represents the number of unique tokens in all subword-level attribute values. Consequently, the vectorization of set $V_c = \{v_{u_1}^c, v_{u_2}^c, \dots, v_{u_i}^c, \dots, v_{u_n}^c\}$ with n users of a social network can be represented as the matrix form, denoted by $H_c = [\vec{v}_{u_1}^c, \vec{v}_{u_2}^c, \dots, \vec{v}_{u_i}^c, \dots, \vec{v}_{u_n}^c]^T$. In this section, the truncated SVD algorithm [47] is applied to implement the dimensionality reduction of H_c . This method is particularly useful for studying sparse matrices efficiently because it does not center the data before computing the singular value decomposition. As a result, H_c is represented by the subword-level feature matrix $H'_c \in \mathbb{R}^{|U| \times d_c}$ with d_c dimensions.

The main goal of feature vectorization is to project two target alignment networks into a common embedding space. There are relatively short distances between the users of aligned pairs. We adopt a three-layer MLP f_1 to realize the transformation. The MLP output is the subword-level attribute features of a user u_i , which is represented as $h_{u_i}^{char}$. ReLU is used as the activation function. Dropout and batch normalization are also utilized to assist the model training. The objective function will be introduced in Section 4.4.

4.2.3. Word-Level Attribute Embedding

The word-level attribute features of users are learned in a similar way. First of all, each user's word-level attribute value $v_{u_i}^w$ is tokenized into a set of unique words. Then, the value $v_{u_i}^w$ can be represented as a bag-of-words vector $\vec{v}_{u_i}^w \in \mathbb{R}^{1 \times \mathcal{M}}$:

$$\vec{v}_{u_i}^w = [\text{count}_1^i, \text{count}_2^i, \dots, \text{count}_w^i, \dots, \text{count}_{\mathcal{M}}^i] \quad (6)$$

where count_w^i is the count number of the word w in $v_{u_i}^w$, and \mathcal{M} is the size of the vocabulary. The vectorization analysis of total n users $V_c = \{v_{u_1}^w, v_{u_2}^w, \dots, v_{u_i}^w, \dots, v_{u_n}^w\}$ in a social network can be obtained as the vector form: $H_w = [\vec{v}_{u_1}^w, \vec{v}_{u_2}^w, \dots, \vec{v}_{u_i}^w, \dots, \vec{v}_{u_n}^w]^T$. We also apply dimensionality reduction on H_w to obtain the word-level feature matrix $H'_w \in \mathbb{R}^{|U| \times d_w}$ with d_w dimensions. On top of that, a well-designed MLP f_2 that has a similar architecture of f_1 is used to implement word-level feature representation. The final MLP output is considered as the word-level attribute features of a user, which are represented as $h_{u_i}^{word}$.

4.3. Attention-Based Structure Embedding

This section attempts to show the process of network structure embedding that appears in the model JARUA. The objective is to map the structure information of two target alignment social networks into a unified vector space, which allows more accurate user alignment compared with different metric spaces. There is a natural intuition: if two users have the same structural role, e.g., similar neighborhoods, they are more likely to be the same person.

The structural characteristics of a social network can be derived from user relations. Mapping the relations to an embedding vector such that the user pairs with similar structural roles in different networks are closely located in the embedding space is the core problem of leveraging social network structure information. We address this problem with a two-phase operation: extending the input networks' relations and applying a graph attention network on the combined network. This study discusses node alignment in the scene of multiple social networks. There can be various types of user relationships between users in social networks, e.g., friend, hostility, following fans, and participation. Some of them are directed and partly undirected. Hence, we simply ignore the direction of a directed relation, i.e., relation (u_i, u_j) is equivalent to (u_j, u_i) , for generality.

4.3.1. Relation Extension

Some actual relations in a social network may not be observed because they are not explicitly built or obtained by the crawlers. The lack of a sufficient number of social relations on UA issues can lead to an inefficient model. Typically, two related social networks have a small number of distinctly aligned nodes that are called pre-aligned users. The existing pre-aligned user pairs have contributed to the relation extension. This section describes the extension of the user relations between networks through pre-aligned user pairs.

Suppose that there are a couple of pre-aligned users from target networks G^X and G^Y . If a user has a connection edge in network G^Y , the corresponding user in network G^X should also have this edge. Based on such an observation, we add a relation to a user in network G^X if their counterpart in network G^Y has this relation. Formally, given networks

G^X and G^Y with a set of pre-aligned pairs P , the extended relation set \tilde{R}^X of G^X can be estimated by:

$$\tilde{R}^X = R^X \cup \{(u_i^X, u_j^X) | (u_i^Y, u_j^Y) \in R^Y, (u_i^X, u_j^X) \notin R^X, (u_i^X, u_i^Y) \in P, (u_j^X, u_j^Y) \in P\} \quad (7)$$

Similarly, the relation set of the target network G^Y is extended into \tilde{R}^Y .

4.3.2. Graph Attention Network

GAT has been proven to be successful in many tasks in natural language processing and image processing. This study introduces the mechanism of GAT to learn alignment-oriented embeddings for linking social networks. The aim of GAT is to calculate each user's hidden representations in two networks by their user neighbors. GAT follows a self-attention strategy in its learning process.

First, the embedding $h_{u_i} \in \mathbb{R}^{1 \times d_s}$ of each user u_i with dimension d_s is randomly initialized. Second, we average the user u_i with its neighbors as:

$$h_{u_i}^{in} = \frac{1}{|\mathbb{N}_i| + 1} \sum_{u_j \in \mathbb{N}_i \cup u_i} h_{u_j} \quad (8)$$

where \mathbb{N}_i is the set of u_i 's neighboring users. The attention coefficients can be calculated by:

$$e_{ij} = \vec{v}^T [h_{u_i}^{in} \| h_{u_j}^{in}] \quad (9)$$

where e_{ij} represents the weighted importance of neighboring u_j to u_i , notation $\|$ indicates the concatenate operation, and $\vec{v} \in \mathbb{R}^{1 \times 2d_s}$ represents the shared attention weight vector. Following [48], we do not adopt the transformation matrix W for the input feature in Equation (9), which is different from the original GAT. Third, the adjacent users of u_i are normalized using a softmax function and a LeakyReLU nonlinearity with a negative input slope of 0.2. Such normalization makes the coefficients between different users easy to compare, which is calculated as:

$$\alpha_{ij} = \text{Softmax}(e_{ij}) = \frac{\exp(\text{LeakyReLU}(e_{ij}))}{\sum_{k \in \mathbb{N}_i} \exp(\text{LeakyReLU}(e_{ik}))} \quad (10)$$

A nonlinear function $\text{ReLU}(\cdot) = \max(\cdot, 0)$ is applied to the combination of participating neighbors. The operation generates the output features $h_{u_i}^{out}$ of each user as:

$$h_{u_i}^{out} = \text{ReLU} \left(\sum_{u_j \in \mathbb{N}_i} \alpha_{ij} h_{u_j}^{in} \right) \quad (11)$$

This work also adopts the multi-head mechanism to realize a stable training process. Specifically, a total of Z independent attention heads execute the transformation of Equation (11). Consequently, the averaged features yield the following output feature:

$$h_{u_i}^{out} = \text{ReLU} \left(\frac{1}{Z} \sum_{z=1}^Z \left[\sum_{u_j \in \mathbb{N}_i} \alpha_{ij}^z h_{u_j}^{in} \right] \right) \quad (12)$$

where z is the indicator of an attention head and α_{ij}^z represents the attention coefficient of the z -th attention head. We further expand the attention mechanism to multi-hop neighboring level information by attaching more layers, forming a more global-aware representation of social networks. Let $h_{u_i}^{out(0)}, \dots, h_{u_i}^{out(l)}$ be the output features of user u_i

from 0-th (input features) to l -th layer. The final output features $h_{u_i}^{struc}$ of user u_i can be obtained by concatenating all the layers together as:

$$h_{u_i}^{struc} = [h_{u_i}^{out(0)} \parallel \dots \parallel h_{u_i}^{out(l)}] \quad (13)$$

4.4. User Alignment

As described in the previous sections, we obtain the attribute features and the structural features of two target social networks. This section combines them to form a more substantial representation of each user as:

$$h_{u_i}^* = [h_{u_i}^{char} \parallel h_{u_i}^{word} \parallel h_{u_i}^{struc}] \quad (14)$$

UA issues can be solved by merely comparing the similarity between two users' combined features in a joint vector space. The objective model can find the potentially aligned user pairs from a network by searching for the counterpart network's user's nearest neighbors. This paper adopts Manhattan distance as the similarity measure of users. The similarity between $u_i \in G^X$ and $u_j \in G^Y$ of two networks in the joint vector space is calculated by:

$$\text{sim}(u_i^X, u_j^Y) = |h_{u_i}^{*X} - h_{u_j}^{*Y}| \quad (15)$$

All similar users in G^Y should be calculated using the same method to find the user u_i 's counterpart in G^X . The nearest one is chosen as the u_i 's equivalent. Then, we adopt the following margin-based objective function to ensure the balance between positive and negative samples and guarantee the lower scores for positive ones, i.e.,

$$\mathcal{L} = \sum_{(u_i, u_j) \in P} \{ [\text{sim}(u_i, u_j) - \text{sim}(u_i - e'_j) + \gamma]_+ + [\text{sim}(u_i, u_j) - \text{sim}(e'_i - u_j) + \gamma]_+ \} \quad (16)$$

where notation $[\cdot]_+$ represents the maximum between the parameter ' \cdot ' and the number 0, i.e., $\max(\cdot, 0)$. The symbol γ is a hyperparameter of the margin. e'_i and e'_j denote the negative counterpart of u_i and u_j , respectively. In this work, all users in the target alignment network are randomly selected as negative counterparts. Adam [49] is adopted to minimize the objective function.

4.5. Iterative Training

Iterative training is one of the more practical ways of expanding training data, which provides a method to maximize the use of labeled data on UA issues. The iterative training algorithm applied in this study consists of two steps:

- (1) Train the model with a set of initially labeled data, i.e., pre-linked user pairs.
- (2) Predict the new auto-labeled data iteratively and add them to the next iteration until no new one can be populated.

The quality of the newly labeled data generated in the iterative training is another potential problem. We assess the populated data using the following two measures. First, link users on both sides of a predicted pair should satisfy the nearest neighbor (NN) constraint [48], i.e., each user is the NN of the other. In most cases, the NN relation between two users is asymmetrical. In other words, although a user u_i in G^X is the most similar to u_j in G^Y , there might be another user in G^Y that is closer to u_i . Therefore, user pairs are considered newly aligned pairs only if they are mutually nearest neighbors of each other. Second, the method mentioned above suffers from a limitation. Far-apart user pairs may still satisfy the NN constraint if there exists no user closer to each other. Because higher similarity of potential pairs indicates higher prediction precision, we only fill the user pairs whose similarity is greater than a certain threshold τ into the aligned user pair

set L . This work takes cosine distance as the similarity measure for iterative training. A detailed procedure is given in Algorithm 1.

Algorithm 1: Iterative procedure of JARUA.

Input: Two networks to be aligned $G^X = (U^X, R^X, A^X)$ and $G^Y = (U^Y, R^Y, A^Y)$, a set P of pre-aligned user pairs, $U'^X \subseteq U^X$ and $U'^Y \subseteq U^Y$ represent candidate user sets that does not exist in P , respectively.

Output: The updated set P of model pre-aligned user pairs.

```

1 repeat
2   Reinitialize the model;
3   Train the model based on  $G^X$ ,  $G^Y$  and  $P$ ;
4    $P' \leftarrow \emptyset$ ;
5   for  $u_i \in U'^X$  do
6      $u'_i \leftarrow NN(u_i, U'^Y)$ ;
7     if  $NN(u'_i, U'^X) = u_i$  and  $sim(u_i, u'_i) > \tau$  then
8        $P' \leftarrow add((u_i, u'_i))$ ;
9        $U'^X \leftarrow U'^X - u_i$ ;
10       $U'^Y \leftarrow U'^Y - u'_i$ ;
11    $P \leftarrow P \cup P'$ ;
12 until  $P' = \emptyset$ ;

```

Algorithm 1 takes the candidate user sets U'^X and U'^Y whose elements do not belong to pre-linked pairs as the input. The algorithm output is the extended pre-linked set P with more desired links, which terminates when no newly linked user pairs can be discovered, i.e., $P' \in \emptyset$ (line 12 in Algorithm 1). First, it is required to search for the nearest neighbor member $u'_i \in U'^Y$ of each user $u_i \in U'^X$ by a constraint $u'_i \leftarrow NN(u_i, U'^Y)$ (line 6 in Algorithm 1). Second, the algorithm determines whether the new link pair (u_i, u'_i) is reasonable or not by the reverse judgment of nearest neighbor $NN(u'_i, U'^X) = u_i$ and the similarity threshold classification according to parameter τ (line 7 in Algorithm 1). Finally, all new user pairs that satisfy the linked pair criteria are stored in set P' for updating the set P of pre-aligned user pairs (lines 8–11 in Algorithm 1).

5. Experiments

Aiming to answer the above-mentioned three questions, this section mainly conducts experiments on two real-world datasets introduced in [7] to evaluate our proposed method. All source codes are publicly available online (<https://github.com/ChenBaiyang/JARUA>, accessed on 4 November 2022).

5.1. Data Sets

The data sets include a pair of social networks and another pair of academic coauthor networks. Their statistics are listed in Table 2.

Table 2. The statistics of our data sets.

Dataset	Network	#Users	#Relations	Min. Degree	Ave. Degree	Max. Degree	Ave. Coeff.	#Aligned Pairs
Social networks	Weibo	9714	117,218	2	12.1	607	0.112	1397
	Douban	9526	120,245	2	12.6	608	0.101	
Coauthor networks	DBLP17	9086	51,700	2	5.7	144	0.280	2832
	DBLP19	9325	47,775	2	5.1	138	0.322	

Social networks : This data set contains two popular Chinese social networks: Weibo (Sina Weibo) and Douban denoted as WD. Some Douban users have revealed their own Weibo accounts on their homepages. Thus, these accounts are taken as pre-matched users

for the alignment task. The two social networks enjoy typical social network characteristics, e.g., approximate power-law degree distribution (Figure 3) and high aggregation coefficient.

Coauthor networks: DBLP is a computer science coauthorship network where two authors are linked if they have published at least one paper together. Every author on DBLP is assigned a unique key, which is viewed as the ground truth for UA. DBLP networks also enjoy social network characteristics of approximate power-law degree distribution (Figure 3) and high aggregation coefficient.

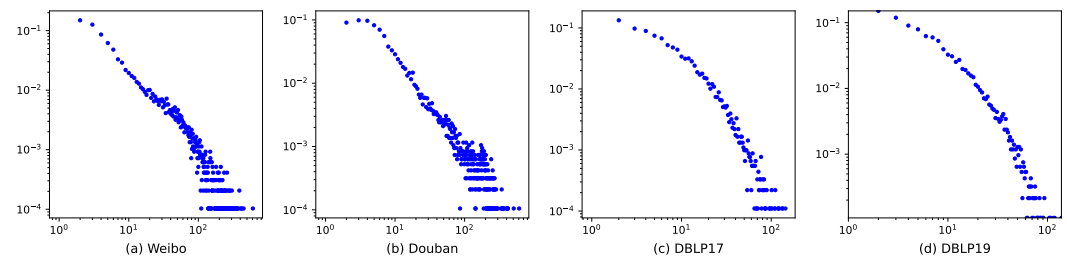


Figure 3. The degree distributions of our experimental data sets.

5.2. Experimental Settings

5.2.1. Comparison Methods

To comprehensively evaluate the comparative performance of our proposed method, we selected the following five models as baselines.

- **IONE** [38]: A semi-supervised method based on network structures, in which users' relations are represented as input/output context vectors.
- **REGAL** [13]: An unsupervised network alignment method that incorporates both network structures and attributes. In this experiment, TF-IDF is utilized to represent node attributes.
- **ABNE** [25]: A graph attention-based UA model, which exploits the network structure by modeling the contribution probabilities in users' relationships.
- **TADW** [50]: An attributed network embedding model. In our experiment, an MLP is employed to map source embeddings to target space for the UA task.
- **PALE** [22]: A supervised method that adopts structural regularities of networks for user alignment. Herein, LINE [51] is taken as a substitute network embedding method for PALE.
- **CENALP** [52]: An embedding model that jointly performs link prediction and network alignment using the skip-gram framework. The weight parameter c is fine-tuned from 0.1 to 0.9 with a step size 0.1 for optimal results.
- **Grad-Align** [53]: A semi-supervised model that gradually predicts aligned node pairs by node consistency and prior information. The gradual matching function is set to a base-2 exponential function.
- **Grad-Align+** [54]: An extension of Grad-Align that use attribute augmentation for UA.

Variants of JARUA: In order to study the effectiveness of the model components of JARUA, we also examine three variants of JARUA, including:

- **JARUA-a**, which ignores the components of structure embedding. Attributes are adopted as the only features to perform the UA task.
- **JARUA-s**, which takes only network structural features for user alignment, ignoring attribute features.
- **JARUA-v**, which directly combines the attribute embedding component and structure embedding component to align users without iterative training.
- **JARUA-t**, which replaces the multi-granularity attribute embedding module of JARUA with TF-IDF vectors to represent user attributes.

All comparison models described above can be divided into three experimental control groups by the features they adopted.

- (1) Structure based: IONE, ABNE, PALE, and JARUA-s.
- (2) Attribute based: JARUA-a.
- (3) Combination of structure and attributes: REGAL, TADW, CENALP, Grad-Align, Grad-Align+, JARUA-v, JARUA-t, and JARUA.

5.2.2. Evaluation Metric

Following [23,42,43], we select Hit-Precision, a popular evaluation metric for the UA task, to evaluate the comparison methods. Hit-Precision computes the top- k candidates in the target network for users in the source network, defined as:

$$\text{Hit-Precision} = \frac{1}{|\mathcal{T}|} \sum_{x_i \in \mathcal{T}} \frac{k - (\text{pos}(x_i) - 1)}{k} \quad (17)$$

where \mathcal{T} is the set of tested user pairs, and $\text{pos}(x_i)$ returns the rank position of user x_i 's corresponding user in the output list of candidates. We report Hit-Precision scores based on $k = 3$ for all the experiments unless otherwise stated.

5.2.3. Implementation Details

Username in WD are taken as subword-level attributes, and the Chinese characters are converted into phonetic alphabets. Locations and posts are treated as word-level attributes. Similarly, authors' names in DBLPs are selected as subword-level attributes. Affiliations and the titles of authors' publications are treated as word-level attributes. To evaluate the model's robustness to noise, this work randomly deletes some attributes with the probability $p = 0.2$.

Without loss of generality, the same settings for experimental models on both data sets are applied unless otherwise stated. The hidden dimensions for subword-level attributes, word-level attributes, and nodes are the same as $D = d_c = d_w = d_s = 100$. The margin parameter γ , attention heads Z , GAT layer's depth l and learning rate of Adam are set as 3, 2, 3, and 0.005, respectively. We empirically set all the dropout rates to 0.3 except the attribute embedding component on DBLP, which is set to 0.5. The main reason for this is that we found it much easier to obtain over-fitting. In the iterative training process, the threshold τ is fixed at 0.8. For the comparison models, their parameters are following the defaults in their paper.

We randomly assigned 30% of the pre-aligned user pairs as training data, while the remaining 70% was used for testing. Each experiment instance was run ten times independently and their average scores are considered the final results.

5.3. Experimental Results

5.3.1. Comparative Performances

The methods in our experiment can be divided into three categories. The first group is purely based on network structure, including IONE, ABNE, PALE, and JARUA-s. The second group is based on user attributes. Here, we have only JARUA-a. The third group integrates attributes and network structure, including REGAL, TADW, CENALP, Grad-Align, Grad-Align+, JARUA-v, JARUA-t, and JARUA. Table 3 summarizes the overall experimental performances.

All four methods in the first group perform better on DBLP than on WD. The reason for this is that the WD structure information is highly noisy and incomplete, and thus cannot provide effective information for recognizing users. For example, IONE enjoys higher Hit-Precision ($k = 3$) by 33.4% and ABNE by 33.8%. The variant model JARUA-s beat the other three models by at least 11.2% on DBLP and 8.4% on WD, demonstrating the superiority of JARUA in capturing the characteristics of the network structure.

Table 3. Hit-Precision scores of all comparison methods.

Method	Weibo-Douban			DBLP17-DBLP19		
	k = 1	k = 3	k = 5	k = 1	k = 3	k = 5
IONE	0.022	0.033	0.042	0.282	0.367	0.418
PALE	0.012	0.021	0.029	0.062	0.104	0.136
ABNE	0.089	0.113	0.128	0.362	0.451	0.502
REGAL	0.016	0.030	0.041	0.411	0.436	0.452
TADW	0.004	0.006	0.008	0.241	0.337	0.398
CENALP	0.076	0.076	0.076	0.175	0.175	0.176
Grad-Align	0.300	0.316	0.329	0.306	0.335	0.353
Grad-Align+	0.299	0.302	0.305	0.288	0.304	0.316
JARUA-s	0.156	0.197	0.220	0.467	0.563	0.615
JARUA-a	0.368	0.401	0.419	0.696	0.723	0.738
JARUA-v	0.383	0.418	0.436	0.740	0.767	0.782
JARUA-t	0.159	0.196	0.221	0.785	0.812	0.829
JARUA	0.416	0.448	0.464	0.821	0.845	0.858

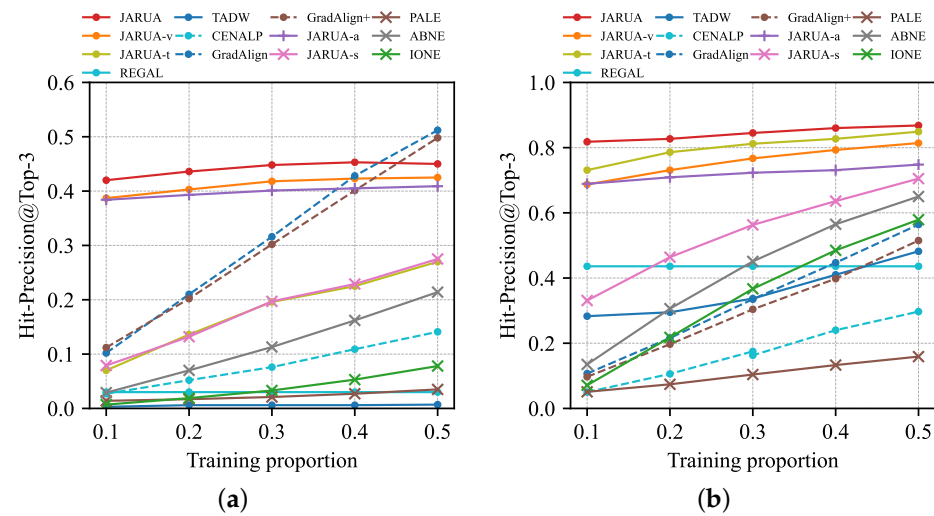
JARUA-a in the second group achieves Hit-Precision scores of 0.401 and 0.723 on WD and DBLP, respectively. The latter is significantly higher than the former. This result is because the latter's attributes are much richer and cleaner, while the attributes on WD, especially users' blogs and posts, fluctuate greatly in the content, number of posts, and publishing frequency. Besides, it can be observed that JARUA-a performs significantly better than the structure-based models. Its score beats other baselines in the first group by at least 29.8% and 27.2% on WD and DBLP, respectively. This is mainly the result of the plentiful attribute information of the data sets.

In the third group, the methods can incorporate both attribute and structure information for UA. JARUA-v performs considerably better than the other five comparison methods, e.g., 10.2% higher than Grad-Align on WD and 33.1% higher than REGAL on DBLP. This is attributed to the differentiation and utilization of multi-type user attributes. Although the others can make use of attributes, they do not distinguish different attribute types. Username plays the strongest role among all attributes on WD, while other attributes (location and posts) are fairly diverse and noisy. Thus, subword-level attributes dominate the model performance.

5.3.2. The Effect of Training Proportions

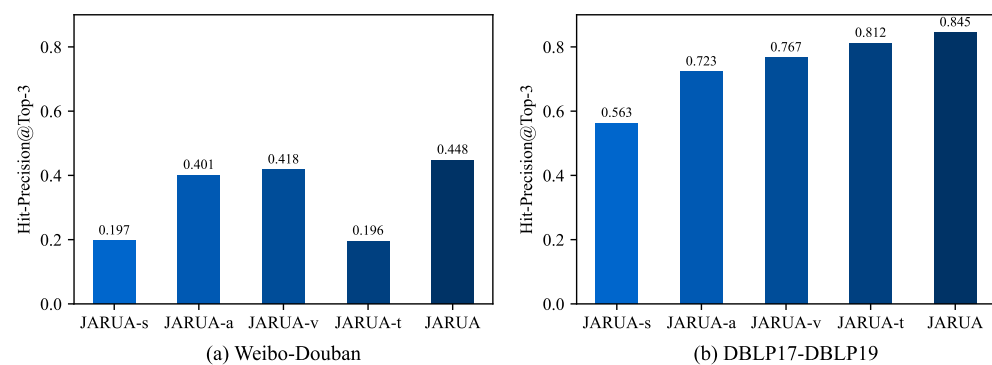
This section evaluates the model performances concerning the proportion of training data. We set the training proportion to vary from 10% to 50% with a step size of 10%. Figure 4a illustrates the change of Hit-Precision of JARUA on social networks. The presented results are improved with the increase in proportion, following our expectations. JARUA outperforms all comparison methods when the training proportion is less than 50%, while Grad-Align and Grad-Align+ achieve the best scores when training data are 50%. So, the proposed method enjoys higher performances in semi-supervised scenarios.

Figure 4b depicts the performances of JARUA on coauthor networks. With the increasing training proportion, all methods (except REGAL) present similar upward trend curves. ABNE and IONE increase faster than JARUA-s, TADW, and PALE. However, JARUA achieves the best final results, e.g., JARUA beats ABNE and IONE by 21.8% and 28.9%. We can also recognize that JARUA performed encouragingly when using a small number of pre-aligned users as training data. For example, when the training proportion is 10%, JARUA scores 42.0% and 81.8% in terms of Hit-Precision on WD and DBLPs, respectively. Therefore, JARUA is expected to be well adapted to annotate constrained scenarios.



5.3.3. The Contribution of Model Components

As demonstrated in Figure 4a,b, JARUA constantly outperforms all four variants across all training proportions. Here, we take a 30% training proportion as an example to discuss the contributions of JARUA components. Figure 5 shows the performances of the variants of JARUA. Clearly, JARUA performs better than JARUA-s by 25.1% on WD and 28.2% on DBLP, which can be attributed to incorporating attributes information. JARUA beats JARUA-a by 4.7% on WD and 12.2% on DBLP, suggesting that utilizing network structural information benefits the UA tasks. However, this effect is not so strong as attribute information in our data sets. Thus, JARUA is quite promising when attribute information is sufficient. It is found that JARUA achieves Hit-Precision values 3.0% higher than JARUA-v on WD and 7.8% higher on DBLP. This suggests that the iterative training algorithm significantly improves the model performances. Moreover, JARUA beats JARUA-t by 25.6% on WD and 3.3% on DBLP. This demonstrates that the proposed multi-granularity attribute embedding performs well in modeling multi-type text attributes.



5.3.4. Parameter Analysis

This section studies the parameter sensitivity of our methods on four primary hyperparameters: (1) the number of attention heads Z , (2) GAT layer’s depth l , (3) the margin parameter γ of the objective function, and (4) the feature dimension D . We adjust the former three parameters by increasing from 1 to 4, and the feature dimension D changes from 50 to 200 with a step size of 50. Starting with the default setting stated in Section 5.2, we only change one hyperparameter value each time. The experiments are carried out on both data

sets, and the results are listed in Table 4. The proposed approach constantly maintains a stable performance with the changes of all the hyperparameters, which implies that our method is robust to hyperparameter tuning.

Table 4. Parameter sensitivity experiment results.

Parameters	Weibo-Douban			DBLP17-DBLP19		
	k = 1	k = 3	k = 5	k = 1	k = 3	k = 5
$Z = 1$	0.385	0.418	0.436	0.743	0.770	0.785
$Z = 2$	0.383	0.418	0.436	0.740	0.767	0.782
$Z = 3$	0.385	0.417	0.435	0.739	0.767	0.782
$Z = 4$	0.384	0.415	0.433	0.740	0.769	0.783
Range	0.002	0.003	0.004	0.004	0.003	0.003
$l = 1$	0.387	0.419	0.437	0.732	0.762	0.777
$l = 2$	0.389	0.419	0.436	0.743	0.770	0.784
$l = 3$	0.383	0.418	0.436	0.740	0.767	0.782
$l = 4$	0.382	0.414	0.432	0.741	0.767	0.782
Range	0.007	0.005	0.005	0.011	0.008	0.008
$\gamma = 1$	0.382	0.412	0.429	0.712	0.740	0.756
$\gamma = 2$	0.381	0.415	0.433	0.727	0.755	0.771
$\gamma = 3$	0.383	0.418	0.436	0.740	0.767	0.782
$\gamma = 4$	0.387	0.419	0.438	0.750	0.777	0.791
Range	0.006	0.007	0.009	0.038	0.037	0.036
$D = 50$	0.384	0.414	0.431	0.734	0.770	0.789
$D = 100$	0.383	0.418	0.436	0.740	0.767	0.782
$D = 150$	0.375	0.411	0.430	0.731	0.757	0.770
$D = 200$	0.369	0.400	0.417	0.719	0.742	0.755
Range	0.015	0.018	0.019	0.021	0.027	0.034

5.3.5. Discussions

User attributes of social networks play a significant role in detecting linked user pairs in the UA problem when pre-labeled user text data are sufficient. The attribute data with different types may lead to model performance differences that can not be ignored. UA tasks are not sensitive to a purely network-structured approach because the structure samples, e.g., friend relationships and comments, are usually implicit or partially fabricated. However, the network structure information can be utilized as compensation to improve the accuracy of attribute models. Affirming the balance of feature combination between user attributes and network structures is an excellent method to enhance UA model performance.

The UA models' accuracy is limited by the percentage of pre-aligned (labeled) user pairs that appear across the networks. Proposing unlabeled data mining opinions to enlarge the labeled data set is a playful process to improve the model's performance. Making full use of unlabeled data remains a potential direction in UA issues.

6. Conclusions and Future Work

This paper studies the UA problem and proposes a novel embedding-based model, JARUA, which incorporates attributes and structure features into a unified vector space. JARUA first captures the characteristics of multi-type attributes through multi-granularity attribute embedding. Then, the proposed model adopts a GAT to embed the network structure. A unified objective function optimizes the attribute embedding component and structure embedding component. Finally, JARUA utilizes unlabeled data with an iterative training algorithm with quality filters. The experimental analysis demonstrates that: (1) leveraging both features of multi-type user attributes and network structure

can significantly improve UA performance; (2) the iterative training algorithm developed in JARUA significantly improves model performances; (3) JARUA is expected to be well adapted to annotating constrained scenarios; (4) the proposed approach is robust to hyperparameter tuning.

The insights into model construction gained from this study may be of assistance to cross-network user profile and network integration. Our future work seeks to extend the ideas of JARUA to other promising applications, such as link prediction, cross-network information diffusing, and network dynamics analysis.

Author Contributions: Conceptualization, M.Y. and B.C.; methodology, M.Y.; software, B.C.; validation, X.C.; formal analysis, M.Y.; investigation, M.Y. and B.C.; resources and data curation, B.C.; writing—original draft preparation, M.Y.; writing—review and editing, B.C. and X.C.; visualization, B.C.; supervision, X.C.; project administration, X.C.; funding acquisition, X.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Science and Technology Program of Sichuan Province (Grant No. 2023ZDYF2732) and National Natural Science Foundation of China (Grant No. 61902324).

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: <https://github.com/ChenBaiyang/JARUA>, accessed on 4 November 2022.

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- Shu, K.; Wang, S.; Tang, J.; Zafarani, R.; Liu, H. User Identity Linkage across Online Social Networks: A Review. *Sigkdd Explor. Newsl.* **2017**, *18*, 5–17. [\[CrossRef\]](#)
- Huang, S.; Zhang, J.; Wang, L.; Hua, X.S. Social Friend Recommendation Based on Multiple Network Correlation. *IEEE Trans. Multimed.* **2016**, *18*, 287–299. [\[CrossRef\]](#)
- Zhang, J.; Yu, P.S.; Zhou, Z.H. Meta-path based multi-network collective link prediction. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, 24–27 August 2014; pp. 1286–1295. [\[CrossRef\]](#)
- Zafarani, R.; Liu, H. Users joining multiple sites: Distributions and patterns. In Proceedings of the 8th International Conference on Weblogs and Social Media, Ann Arbor, MI, USA, 1–4 June 2014; pp. 635–638.
- Zafarani, R.; Liu, H. Users joining multiple sites: Friendship and popularity variations across sites. *Inf. Fusion* **2016**, *28*, 83–89. [\[CrossRef\]](#)
- Zhang, J.; Chen, B.; Wang, X.; Song, G.; Chen, H.; Li, C.; Jin, F.; Zhang, Y. MEgo2Vec: Embedding matched ego networks for user alignment across social networks. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino, Italy, 22–26 October 2018; pp. 327–336. [\[CrossRef\]](#)
- Chen, B.; Chen, X. MAUIL: Multi-level attribute embedding for semi-supervised user identity linkage. *Inf. Sci.* **2022**, *593*, 527–545. [\[CrossRef\]](#)
- Malhotra, A.; Totti, L.; Meira, W., Jr.; Kumaraguru, P.; Almeida, V. Studying User Footprints in Different Online Social Networks. In Proceedings of the International Conference on Advances in Social Networks Analysis and Mining, Istanbul, Turkey, 26–29 August 2012; pp. 1065–1070. [\[CrossRef\]](#)
- Acquisti, A.; Gross, R.; Stutzman, F. Face Recognition and Privacy in the Age of Augmented Reality. *J. Priv. Confid.* **2014**, *6*, 638. [\[CrossRef\]](#)
- Feng, J.; Zhang, M.; Wang, H.; Yang, Z.; Zhang, C.; Li, Y.; Jin, D. DPLink: User identity linkage via deep neural network from heterogeneous mobility data. In Proceedings of the World Wide Web Conference, San Francisco, CA, USA, 13–17 May 2019; pp. 459–469. [\[CrossRef\]](#)
- Zhang, J.; Kong, X.; Yu, P.S. Transferring Heterogeneous Links across Location-Based Social Networks. In Proceedings of the 7th ACM international Conference on Web Search and Data Mining, New York, NY, USA, 24–28 February 2014; pp. 303–312.
- Xiaolin, C.; Xuemeng, S.; Siwei, C.; Tian, G.; Zhiyong, C.; Liqiang, N. User Identity Linkage Across Social Media via Attentive Time-Aware User Modeling. *IEEE Trans. Multimed.* **2021**, *23*, 3957–3967. [\[CrossRef\]](#)
- Heimann, M.; Safavi, T.; Shen, H.; Koutra, D. REGAL: Representation Learning-based Graph Alignment. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino, Italy, 22–26 October 2018; pp. 117–126. [\[CrossRef\]](#)
- Lu, M.; Dai, Y.; Zhang, Z. Social network alignment: A bi-layer graph attention neural networks based method. *Appl. Intell.* **2022**, *52*, 16310–16333. [\[CrossRef\]](#)
- Zhong, Z.; Cao, Y.; Guo, M.; Nie, Z. CoLink: An unsupervised framework for user identity linkage. In Proceedings of the 32nd AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 5714–5721.

16. Zhang, Y.; Tang, J.; Yang, Z.; Pei, J.; Yu, P.S. COSNET: Connecting Heterogeneous Social Networks with Local and Global Consistency. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, 10–13 August 2015; pp. 1485–1494. [\[CrossRef\]](#)
17. Velickovi, P.; Casanova, A.; Lio, P.; Cucurull, G.; Romero, A.; Bengio, Y. Graph attention networks. In Proceedings of the 6th International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
18. Zafarani, R.; Liu, H. Connecting Corresponding Identities across Communities. In Proceedings of the 3rd International Conference on Weblogs and Social Media, San Jose, CA, USA, 17–20 May 2009; pp. 354–357.
19. Vosecky, J.; Hong, D.; Shen, V.Y. User identification across multiple social networks. In Proceedings of the 1st International Conference on Networked Digital Technologies, Ostrava, Czech Republic, 28–31 July 2009; pp. 360–365. [\[CrossRef\]](#)
20. Kong, X.; Zhang, J.; Yu, P.S. Inferring anchor links across multiple heterogeneous social networks. In Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, San Francisco, CA, USA, 27 October–1 November 2013; pp. 179–188. [\[CrossRef\]](#)
21. Iofciu, T.; Fankhauser, P.; Abel, F.; Bischoff, K. Identifying Users Across Social Tagging Systems. In Proceedings of the 5th International Conference on Weblogs and Social Media, Barcelona, Spain, 17–21 July 2011; pp. 522–525.
22. Man, T.; Shen, H.; Liu, S.; Jin, X.; Cheng, X. Predict anchor links across social networks via an embedding approach. In Proceedings of the 25th International Joint Conference on Artificial Intelligence, New York, NY, USA, 9–15 July 2016; pp. 1823–1829.
23. Mu, X.; Zhu, F.; Lim, E.P.; Xiao, J.; Wang, J.; Zhou, Z.H. User identity linkage by latent user space modelling. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1775–1784. [\[CrossRef\]](#)
24. Liu, S.; Wang, S.; Zhu, F. Structured Learning from Heterogeneous Behavior for Social Identity Linkage. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 2005–2019. [\[CrossRef\]](#)
25. Liu, L.; Zhang, Y.M.; Fu, S.; Zhong, F.J.; Hu, J.; Zhang, P. ABNE: An Attention-Based Network Embedding for User Alignment Across Social Networks. *IEEE Access* **2019**, *7*, 23595–23605. [\[CrossRef\]](#)
26. Qiao, Y.; Wu, Y.; Duo, F.; Lin, W.; Yang, J. Siamese Neural Networks for User Identity Linkage Through Web Browsing. *IEEE Trans. Neural Netw. Learn. Syst.* **2019**, *31*, 2741–2751. [\[CrossRef\]](#)
27. Li, Y.; Su, Z.; Yang, J.; Gao, C. Exploiting similarities of user friendship networks across social networks for user identification. *Inf. Sci.* **2020**, *506*, 78–98. [\[CrossRef\]](#)
28. Fu, S.; Wang, G.; Xia, S.; Liu, L. Deep multi-granularity graph embedding for user identity linkage across social networks. *Knowl.-Based Syst.* **2020**, *193*, 105301. [\[CrossRef\]](#)
29. Li, X.X.; Cao, Y.A.; Li, Q.; Shang, Y.M.; Li, Y.X.; Liu, Y.B.; Xu, G.D. RLINK: Deep reinforcement learning for user identity linkage. *World Wide Web* **2021**, *24*, 85–103. [\[CrossRef\]](#)
30. Liu, J.; Zhang, F.; Song, X.; Song, Y.I.; Lin, C.Y.; Hon, H.W. What's in a name? An unsupervised approach to link users across communities. In Proceedings of the 6th ACM International Conference on Web Search and Data Mining, Rome, Italy, 4–8 February 2013; pp. 495–504. [\[CrossRef\]](#)
31. Li, C.; Wang, S.; Yu, P.S.; Zheng, L.; Zhang, X.; Li, Z.; Liang, Y. Distribution distance minimization for unsupervised user identity linkage. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Torino, Italy, 22–26 October 2018; pp. 447–456. [\[CrossRef\]](#)
32. Lacoste-Julien, S.; Palla, K.; Davies, A.; Kasneci, G.; Graepel, T.; Ghahramani, Z. SiGMA: Simple greedy matching for aligning large knowledge bases. In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, IL, USA, 11–14 August 2013; pp. 572–580. [\[CrossRef\]](#)
33. Nie, Y.P.; Jia, Y.; Li, S.D.; Zhu, X.; Li, A.P.; Zhou, B. Identifying users across social networks based on dynamic core interests. *Neurocomputing* **2016**, *210*, 107–115. [\[CrossRef\]](#)
34. Riederer, C.; Kim, Y.; Chaintreau, A.; Korula, N.; Lattanzi, S. Linking Users Across Domains with Location Data: Theory and Validation. In Proceedings of the 25th International Conference on World Wide Web, Montréal, QC, Canada, 11–15 May 2016; pp. 707–719. [\[CrossRef\]](#)
35. Zhou, X.; Liang, X.; Du, X.; Zhao, J. Structure Based User Identification across Social Networks. *IEEE Trans. Knowl. Data Eng.* **2018**, *30*, 1178–1191. [\[CrossRef\]](#)
36. Zhou, F.; Wen, Z.J.; Zhong, T.; Trajcevski, G.; Xu, X.; Liu, L.Y. Unsupervised User Identity Linkage via Graph Neural Networks. In Proceedings of the 2020 IEEE Global Communications Conference (GLOBECOM 2020), Taipei, Taiwan, 7–11 December 2020. [\[CrossRef\]](#)
37. Li, Y.; Ji, W.; Gao, X.; Deng, Y.; Dong, W.; Li, D. Matching user accounts with spatio-temporal awareness across social networks. *Inf. Sci.* **2021**, *570*, 1–15. [\[CrossRef\]](#)
38. Liu, L.; Cheung, W.K.; Li, X.; Liao, L. Aligning users across social networks using network embedding. In Proceedings of the 25th International Joint Conference on Artificial Intelligence, New York, NY, USA, 9–15 July 2016; pp. 1774–1780.
39. Tan, S.; Guan, Z.; Cai, D.; Qin, X.; Bu, J.; Chen, C. Mapping users across networks by manifold alignment on hypergraph. In Proceedings of the 28th AAAI Conference on Artificial Intelligence, Québec, QC, Canada, 27–31 July 2014; pp. 159–165.
40. Zhou, X.; Liang, X.; Zhang, H.; Ma, Y. Cross-Platform Identification of Anonymous Identical Users in Multiple Social Media Networks. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 411–424. [\[CrossRef\]](#)

41. Zhao, W.; Tan, S.; Guan, Z.; Zhang, B.; Gong, M.; Cao, Z.; Wang, Q. Learning to Map Social Network Users by Unified Manifold Alignment on Hypergraph. *IEEE Trans. Neural Netw. Learn. Syst.* **2018**, *29*, 5834–5846. [\[CrossRef\]](#)
42. Li, C.; Wang, S.; Wang, Y.; Yu, P.; Liang, Y.; Liu, Y.; Li, Z. Adversarial Learning for Weakly-Supervised Social Network Alignment. In Proceedings of the 33rd AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 996–1003. [\[CrossRef\]](#)
43. Li, C.; Wang, S.; Wang, H.; Liang, Y.; Yu, P.S.; Li, Z.; Wang, W. Partially shared adversarial learning for semi-supervised multi-platform user identity linkage. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, Beijing, China, 3–7 November 2019; pp. 249–258. [\[CrossRef\]](#)
44. Liu, L.; Li, X.; Cheung, W.K.; Liao, L.J. Structural Representation Learning for User Alignment Across Social Networks. *IEEE Trans. Knowl. Data Eng.* **2020**, *32*, 1824–1837. [\[CrossRef\]](#)
45. Liu, X.W.; Chen, Y.J.; Fu, J.M. MFRep: Joint user and employer alignment across heterogeneous social networks. *Neurocomputing* **2020**, *414*, 36–56. [\[CrossRef\]](#)
46. Tata, S.; Patel, J. Estimating the Selectivity of tf-idf based Cosine Similarity Predicates. *SIGMOD Rec.* **2007**, *36*, 7–12. [\[CrossRef\]](#)
47. Halko, N.; Martinsson, P.G.; Tropp, J.A. Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions. *SIAM Rev.* **2009**, *53*, 217–288. [\[CrossRef\]](#)
48. Mao, X.; Wang, W.; Xu, H.; Lan, M.; Wu, Y. MRAEA: An Efficient and Robust Entity Alignment Approach for Cross-lingual Knowledge Graph. In Proceedings of the 13th International Conference on Web Search and Data Mining, Houston, TX, USA, 3–7 February 2020; pp. 420–428. [\[CrossRef\]](#)
49. Kingma, D.P.; Ba, J.L. Adam: A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
50. Yang, C.; Liu, Z.; Zhao, D.; Sun, M.; Chang, E.Y. Network Representation Learning with Rich Text Information. In Proceedings of the 24th International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015; pp. 2111–2117.
51. Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; Mei, Q. LINE: Large-scale Information Network Embedding. In Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 1067–1077. [\[CrossRef\]](#)
52. Du, X.; Yan, J.; Zhang, R.; Zha, H. Cross-Network Skip-Gram Embedding for Joint Network Alignment and Link Prediction. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 1080–1095. [\[CrossRef\]](#)
53. Park, J.D.; Tran, C.; Shin, W.Y.; Cao, X. Grad-Align: Gradual Network Alignment via Graph Neural Networks. In Proceedings of the AAAI Conference on Artificial Intelligence, virtually, 22 February–1 March 2022; Volume 36, pp. 13027–13028. [\[CrossRef\]](#)
54. Park, J.D.; Tran, C.; Shin, W.Y.; Cao, X. Grad-Align+: Empowering Gradual Network Alignment Using Attribute Augmentation. In Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, 17–21 October 2022; CIKM 2022, pp. 4374–4378. [\[CrossRef\]](#)