

# Collective Link Prediction Oriented Network Embedding with Hierarchical Graph Attention

Yizhu Jiao

Shanghai Key Laboratory of Data Science, Shanghai  
Institute for Advanced Communication and Data Science,  
School of Computer Science, Fudan University,  
Shanghai, China, yzjiao18@fudan.edu.cn

Yun Xiong

Shanghai Key Laboratory of Data Science, Shanghai  
Institute for Advanced Communication and Data Science,  
School of Computer Science, Fudan University,  
Shanghai, China, yunx@fudan.edu.cn

Jiawei Zhang

IFM Lab, Department of Computer Science  
Florida State University, Tallahassee, FL, USA  
jiawei@ifmlab.org

Yangyong Zhu

Shanghai Key Laboratory of Data Science, Shanghai  
Institute for Advanced Communication and Data Science,  
School of Computer Science, Fudan University,  
Shanghai, China, yyzhu@fudan.edu.cn

## ABSTRACT

To enjoy more social network services, users nowadays are usually involved in multiple online sites at the same time. Aligned social networks provide more information to alleviate the problem of data insufficiency. In this paper, we target on the collective link prediction problem and aim to predict both the intra-network social links as well as the inter-network anchor links across multiple aligned social networks. It is not an easy task, and the major challenges involve the network characteristic difference problem and different directivity properties of the social and anchor links to be predicted. To address the problem, we propose an application oriented network embedding framework, Hierarchical Graph Attention based Network Embedding (HGANE), for collective link prediction over directed aligned networks. Very different from the conventional general network embedding models, HGANE effectively incorporates the collective link prediction task objectives into consideration. It learns the representations of nodes by aggregating information from both the intra-network neighbors (connected by social links) and inter-network partners (connected by anchor links). What's more, we introduce a hierarchical graph attention mechanism for the intra-network neighbors and inter-network partners respectively, which resolves the network characteristic differences and the link directivity challenges effectively. Extensive experiments have been conducted on real-world aligned networks datasets to demonstrate that our model outperformed the state-of-the-art baseline methods in addressing the collective link prediction problem by a large margin.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](https://permissions.acm.org).

CIKM '19, November 3–7, 2019, Beijing, China  
© 2019 Association for Computing Machinery.  
ACM ISBN 978-1-4503-6976-3/19/11...\$15.00  
<https://doi.org/10.1145/3357384.3357990>

## CCS CONCEPTS

• **Information systems** → **Data mining**; *Social networks*; • **Computing methodologies** → *Knowledge representation and reasoning*;

## KEYWORDS

Multiple Aligned Networks, Collective Link Prediction, Attention, Embedding

## ACM Reference Format:

Yizhu Jiao, Yun Xiong, Jiawei Zhang, and Yangyong Zhu. 2019. Collective Link Prediction Oriented Network Embedding with Hierarchical Graph Attention. In *Proceedings of The 28th ACM International Conference on Information and Knowledge Management (CIKM '19)*. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3357384.3357990>

## 1 INTRODUCTION

Nowadays, online social networks have become very popular and extensively used in our lives. To enjoy more services, it is ubiquitous for users to participate in multiple online social platforms concurrently. For example, users may share photos with Instagram and check the latest news information via Twitter. To simplify the sign up/in process, most social platforms usually allow users to use their existing Twitter/Facebook/Google IDs to create their accounts at these new social sites, which will align different online networks together naturally. Each of these platforms can be represented as a massive network where nodes represent user accounts and intra-network links represent the social relationships among users. Especially, accounts owned by the same user in different networks are defined as anchor nodes [9] and inter-network corresponding relationships between the anchor users are defined as anchor links [9]. Different online networks connected by anchor links are defined as multiple aligned social networks [31].

In recent years, there has been a surge of interest in multi-network analysis. Traditional methods that target on one single network require sufficient information to build effective models.

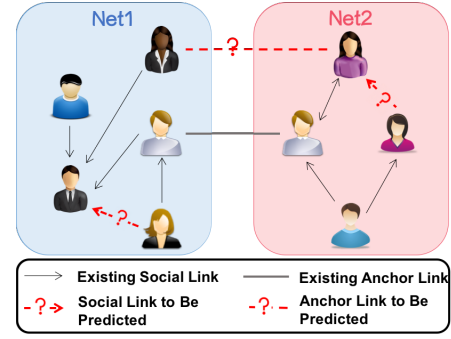
However, as proposed in [25], this assumption can be violated seriously when dealing with the cold start [10] and data sparsity problems. The study of multiple aligned networks provides a direction to alleviate the data insufficiency problem. Some research works propose to transfer information across networks by anchor links to enhance the link prediction results within multiple networks mutually [1, 30, 31]. Besides, many existing works aim at anchor link formation prediction [11, 12, 21]. However, most of these works study either intra-network or inter-network link prediction tasks separately. As discovered in [26], multiple link prediction tasks in the same networks can actually be done simultaneously and enhanced mutually due to their strong correlations.

Predicting multiple kinds of links among users across multiple aligned networks is defined as the collective link prediction problem in [23]. The collective link prediction problem covers several different link formation prediction tasks simultaneously including both the intra-network social link prediction and the inter-network anchor link prediction. It can take advantage of the strong correlations between these different tasks to enhance the prediction performance across these aligned networks synergistically. Figure 1 shows an example of the collective link prediction tasks of two social networks. In the figure, black lines with the arrow indicate existing directed intra-network social links and the gray lines indicate the existing inter-network anchor links instead. These directed/undirected red dashed lines with question marks signify the potential intra-network and inter-network links to be predicted, respectively.

The problem of collective link prediction is worth exploring due to both its importance and novelty. Some existing methods have been introduced to tentatively address the problem [23]. However, these existing methods mostly ignore the contradiction of different characteristics of aligned networks or adopt fixed parameters to control the proportion of information diffused across networks, which usually need to be fine-tuned manually. Besides, these works also fail to consider the connectivity of the links within and across networks.

The collective link prediction problem studied in this paper is also very challenging to solve due to the following reasons:

- **Network characteristic differences:** Since users normally join in different networks for different purposes, each network usually has different characteristics and reveals different aspects of the users. For example, professional relations are established on LinkedIn while personal social-tiers are built in Twitter. Thus, information transferred from other networks may be different from the target network that we want to study. Previous work found information transfer could also deteriorate the performance of intra-network link prediction [31]. Correspondingly, anchor link prediction can be more susceptible as anchor links are directly related to information transferred across networks. Therefore, it is more crucial but challenging to overcome network characteristic difference problem for collective link prediction.
- **Link directivity differences:** The intra-network social links are usually uni-directed from the initiator pointing to the recipient instead. For the users involved in the social



**Figure 1: An example of collective link prediction over multiple aligned networks**

network, the social links pointing to them reflect the objective recognition from the community, whereas that from them to others reflect their personal social interest. Thus, these social relations collaboratively define a unique character in social networks. However, the inter-network anchor links are bi-directed according to the definition. Such different directivity properties on social links and anchor links should be carefully considered in the prediction model.

In this paper, we propose a novel application oriented network embedding framework, namely Hierarchical Graph Attention based Network Embedding (HGANE), to solve the collective link prediction problem over aligned networks. Very different from the conventional general network embedding models, HGANE effectively incorporates the collective link prediction task objectives into consideration. It learns node embeddings in multiple aligned networks by aggregating information from the related nodes, including both the intra-network social neighbors and inter-network anchor partners. What's more, we introduce a hierarchical graph attention mechanism for the intra-network neighbors and inter-network partners respectively, which handles the network characteristic differences and link directivity differences. HGANE balances between the prediction tasks of both the intra-network social link and inter-network anchor link and the learned embedding results can resolve the collective link prediction problem effectively. We conduct detailed empirical evaluations using several real-world datasets and show that our model outperforms other competitive approaches.

We summarize the main contributions of this paper as follows:

- We propose a novel embedding framework to learn the representations of nodes by aggregating information from both the intra-network neighbors (connected by social links) and inter-network partners (connected by anchor links).
- We introduce a hierarchical graph attention mechanism. It includes two levels of attention mechanisms — one at the node level and one at the network level — to resolve the network characteristic differences and link directivity challenges effectively.
- HGANE incorporates the collective link prediction task objectives into consideration and balances between the prediction tasks of the intra-network social link and inter-network anchor link, respectively.

- Extensive experiments are conducted on two real-world aligned social network datasets. The results demonstrate that the proposed model outperforms existing state-of-the-art approaches by a large margin.

## 2 PRELIMINARY

**Definition 1.** (Multiple Aligned Social Networks) : In this paper, we follow the definitions introduced in [9]. Given  $n$  networks  $\{G^{(1)}, \dots, G^{(n)}\}$  with shared users, they can be defined as multiple aligned networks  $\mathcal{G} = ((G^{(1)}, G^{(2)}, \dots, G^{(n)}), (A^{(1,2)}, A^{(1,3)}, \dots, A^{(1,n)}, A^{(2,3)}, \dots, A^{(n-1,n)}))$ , where  $G^{(i)} = (V^{(i)}, E^{(i)})$ ,  $i \in \{1, 2, \dots, n\}$  is a network consisting of nodes and links, and  $A^{(i,j)}$  represents the anchor links between  $G^{(i)}$  and  $G^{(j)}$ .

For two nodes  $v^{(i)} \in V^{(i)}$  and  $v^{(j)} \in V^{(j)}$ , node pair  $(v^{(i)}, v^{(j)}) \in A^{(i,j)}$  iff  $v^{(i)}$  and  $v^{(j)}$  are the accounts of the same user in networks  $G^{(i)}$  and  $G^{(j)}$  respectively.

For two online networks, such as Foursquare and Twitter used in this paper, we can represent them as two aligned social networks  $\mathcal{G} = ((G^{(1)}, G^{(2)}), (A^{(1,2)}))$ , which will be used as an example to illustrate the model. A simple extension of the proposed framework can be applied to multiple aligned networks conveniently.

**Problem Definition :** The collective link prediction problem studied in this paper includes simultaneous prediction of both intra-network social links and inter-network anchor links. Formally, given two aligned networks  $\mathcal{G} = ((G^{(1)}, G^{(2)}), (A^{(1,2)}))$  where both of  $G^{(1)}$  and  $G^{(2)}$  are directed social networks. We can represent all the unknown social links among the nodes in  $G^{(k)}$  as  $U^{(k)} = V^{(k)} \times V^{(k)} \setminus (E^{(k)} \cup \{(u, u)\}_{u \in V^{(k)}})$  where  $k \in \{1, 2\}$ . And the unknown anchor links across  $G^{(1)}$  and  $G^{(2)}$  can be denoted as  $U^{(1,2)} = V^{(1)} \times V^{(2)} \setminus A^{(1,2)}$ . We aim at building a mapping  $f : U^{(1)} \cup U^{(2)} \cup U^{(1,2)} \rightarrow [0, 1]$  to project these intra-network social links and inter-network anchor links to their formation probabilities.

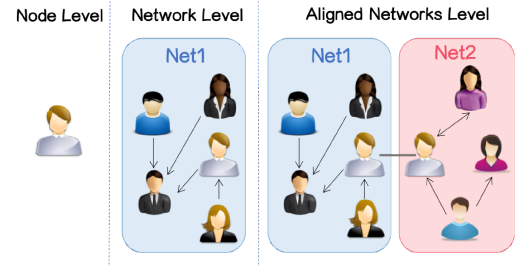
## 3 PROPOSED METHOD

**Table 1: Descriptions of notations in our framework.**

| Notation                    | Description   |
|-----------------------------|---|
| $u_i$                       | Node $i$ in $G^{(1)}$                               |
| $v_j$                       | Node $j$ in $G^{(2)}$                               |
| $\mathbf{u}_i^{in}$         | Initiator feature of $u_i$ in $G^{(1)}$             |
| $\mathbf{v}_j^{in}$         | Initiator feature of $v_j$ in $G^{(2)}$             |
| $\mathbf{u}_i^{re}$         | Recipient feature of $u_i$ in $G^{(1)}$             |
| $\mathbf{v}_j^{re}$         | Recipient feature of $v_j$ in $G^{(2)}$             |
| $N^I(u_i)$                  | Intra-network neighbors of $u_i$ as the initiator   |
| $N^R(u_i)$                  | Intra-network neighbors of $u_i$ as the recipient   |
| $N^A(u_i)$                  | Inter-network anchor partners of $u_i$              |
| $\mathbf{e}^{in}(u_i, u_j)$ | Intra-network initiator attention of $u_i$ to $u_j$ |
| $\mathbf{e}^{in}(u_i, v_j)$ | Inter-network initiator attention of $u_i$ to $v_j$ |
| $\mathbf{e}^{re}(u_i, u_j)$ | Intra-network recipient attention of $u_i$ to $u_j$ |
| $\mathbf{e}^{re}(u_i, v_j)$ | Inter-network recipient attention of $u_i$ to $v_j$ |

In this section, we will introduce the framework HGANE in detail. For the convenience of elaboration, we provide the main notations used through this paper in Table 1. At the beginning, the hierarchical graph attention mechanism will be introduced to handle the problems of network characteristic differences and link directivity challenges. After that, we will propose a novel node embedding method in multiple aligned networks. Finally, we will introduce the application oriented network embedding framework which can resolve the collective link prediction problem effectively.

### 3.1 Hierarchical Graph Attention Mechanism



**Figure 2: Hierarchical structure of multiple aligned networks**

Social networks consist of nodes and social links, while multiple aligned social networks consist of many social networks and anchor links across them. Therefore, multiple aligned social networks have a hierarchical structure, which is illustrated in Figure 2. Besides, for the target node, it is observed that the relevance of different neighbors is different. For the target network, other networks are differentially informative since they have different characteristics. Furthermore, as each node is cooperatively characterized by its neighbor nodes and anchor partners in other networks, their importance is highly dependent on node embeddings.

Therefore, we propose the hierarchical graph attention mechanism in this section. It includes two levels of attention mechanisms — one at the node level and the other at the network level — to make our model pay more or less attention to different neighbor nodes and networks when constructing the node representations. These two levels of attention mechanisms are formally called the *intra-network social attention* and the *inter-network anchor attention*. They are essential to resolve the problems of network characteristic differences and link directivity challenges in the multiple aligned networks. In the following subsections, we will introduce the definitions and motivations of these two attention mechanisms.

#### 3.1.1 Intra-Network Social Attention.

Given two aligned networks  $\mathcal{G} = ((G^{(1)}, G^{(2)}), (A^{(1,2)}))$ , the anchor links are defined to be bi-directed; while the intra-network social links are usually uni-directed from the initiator pointing to the recipient instead. Thus, every node plays these two roles within the social network. Normally, we represent each node  $u_i$  with two vector representations, the initiator feature  $\mathbf{u}_i^{in} \in \mathbb{R}^d$  and the recipient feature  $\mathbf{u}_i^{re} \in \mathbb{R}^d$ , where  $d$  is the feature dimension. The initiator feature  $\mathbf{u}_i^{in}$  represents the characteristic of the node as the

initiator following others while the recipient feature  $\mathbf{u}_i^{re}$  represents the characteristic of the node as the recipient followed by others. By distinguishing the initiator and recipient features for each node, we can lay the foundation for resolving the problem of link directivity challenges effectively.

For the target node  $u_i$ , we define the nodes followed by it as its intra-network recipient neighbors. The set of intra-network recipient neighbors of the initiator  $u_i$  is denoted as  $\mathcal{N}^i(u_i)$ . The node  $u_j \in \mathcal{N}^i(u_i)$  iff  $(u_i, u_j) \in E^{(1)}$ . Similarly, the nodes following  $u_i$  are defined as its intra-network initiator neighbors and the set of these neighbors of the recipient  $u_i$  is denoted as  $\mathcal{N}^r(u_i)$ . The node  $u_j \in \mathcal{N}^r(u_i)$  iff  $(u_j, u_i) \in E^{(1)}$ . Here, based on the two node roles, we introduce *intra-network initiator attention* and *intra-network recipient attention*, to leverage the structural information within the social network.

For intra-network social neighbors, the characteristic of the initiator is relevant to the recipient. The intra-network initiator attention mechanism computes the coefficients to judge the importance of the intra-network recipient neighbor to the target initiator. Here, the concept of intra-network initiator attention mechanism can be represented formally.

**Definition 2.** (Intra-Network Initiator Attention) : For the target node  $u_i$  and its intra-network recipient neighbor  $u_j \in \mathcal{N}^i(u_i)$ , the intra-network initiator attention coefficient of  $u_i$  to  $u_j$  can be given as

$$e^{in}(u_i, u_j) = \sigma\left(\mathbf{a}_{in}^{(1)T} \left[ \mathbf{W}_{in}^{(1)} \mathbf{u}_i^{in} \parallel \mathbf{W}_{re}^{(1)} \mathbf{u}_j^{re} \right]\right),$$

where  $\cdot^T$  represents transposition and  $\parallel$  is the concatenation operation.  $\mathbf{W}_{in}^{(1)} \in \mathbb{R}^{d' \times d}$  and  $\mathbf{W}_{re}^{(1)} \in \mathbb{R}^{d' \times d}$  are the weight matrixes applied to every node as the initiator and the recipient for shared linear transformation in  $G^{(1)}$ .  $\mathbf{a}_{in}^{(1)T} \in \mathbb{R}^{2d'}$  is a weight vector and  $\sigma$  denotes the activation function.  $\mathbf{a}_{re}^{(1)T}$ ,  $\mathbf{a}_{in}^{(1,2)T}$  and  $\mathbf{a}_{re}^{(1,2)T}$  will be used with similar meanings.

Similarly, since the characteristics of the initiator and the recipient are correlative, we can introduce the definition of intra-network initiator attention mechanism formally to obtain the importance of the intra-network initiator neighbor to the target recipient.

**Definition 3.** (Intra-Network Recipient Attention) : For the target node  $u_i$  and its intra-network initiator neighbor  $u_j \in \mathcal{N}^r(u_i)$ , the intra-network recipient attention coefficient of  $u_i$  to  $u_j$  can be given as

$$e^{re}(u_i, u_j) = \sigma\left(\mathbf{a}_{re}^{(1)T} \left[ \mathbf{W}_{re}^{(1)} \mathbf{u}_i^{re} \parallel \mathbf{W}_{in}^{(1)} \mathbf{u}_j^{in} \right]\right),$$

where  $\mathbf{a}_{re}^{(1)T}$  is also a weight vector.

With these two kinds of intra-network attention mechanism, our model can pay more attention on useful information and neglect harmful information within the social networks. It is significant to resolve the problem of directivity challenges effectively and leverage structural information within social networks.

### 3.1.2 Inter-Network Anchor Attention.

Different from the intra-network social attention mechanism which targets at the node level, the inter-network anchor attention is for the network level. The anchor links connecting multiple networks play a crucial role in cross-network information transfer.

However, due to the problem of network characteristic differences, information transferred from other networks could also undermine the learned embeddings of the target network.

To handle this problem, we introduce the inter-network anchor attention mechanism. For the target node, the inter-network anchor attention coefficient to its anchor partners in the other network indicates the importance of information transferred from that network. As mentioned in the last section, each node is represented by the initiator and recipient embeddings. To transfer the directed structural information within networks, two kinds of inter-network anchor attention will be introduced according to these two roles of the nodes.

Different from uni-directed social links, since the anchor nodes reveal the information of the same user from different aspects, it is intuitive that their initiator and recipient features in different networks can be related correspondingly. For the target node and its anchor partner in some network, the inter-network initiator attention coefficient indicates the importance of information from that network to the target node as the initiator. Firstly, we represent the set of the inter-network anchor partners in the other network for the target node  $u_i$  as  $\mathcal{N}^a(u_i)$ . If the node pair  $(u_i, v_j) \in A^{(1,2)}$ ,  $v_j \in \mathcal{N}^a(u_i)$ . And the definition of inter-network initiator attention is introduced as follows.

**Definition 4.** (Inter-Network Initiator Attention): For the target node  $u_i$  and its inter-network anchor partner  $v_j \in \mathcal{N}^a(u_i)$ , the intra-network initiator attention coefficient of  $u_i$  to  $v_j$  as the initiator can be given as

$$e^{in}(u_i, v_j) = \sigma\left(\mathbf{a}_{in}^{(1,2)T} \left[ \mathbf{W}_{in}^{(1)} \mathbf{u}_i^{in} \parallel \mathbf{W}_{in}^{(1,2)} \mathbf{v}_j^{in} \right]\right),$$

where  $\mathbf{W}_{in}^{(1,2)}$  is the weight matrix applied to every anchor node in  $G^{(2)}$  as the initiator while transferring information to  $G^{(1)}$  and  $\mathbf{a}_{in}^{(1,2)T}$  is a weight vector.

Similarly, considering the recipient role of nodes, it is intuitive that the recipient features of anchor partners are related. Based on this, we give the concept of inter-network recipient attention, which denotes the importance of information from that network to the target node as the recipient.

**Definition 5.** (Inter-Network Recipient Attention) : For the target node  $u_i$  and its inter-network anchor partner  $v_j \in \mathcal{N}^a(u_i)$ , the intra-network recipient attention coefficient of  $u_i$  to  $v_j$  as the recipient can be given as

$$e^{re}(u_i, v_j) = \sigma\left(\mathbf{a}_{re}^{(1,2)T} \left[ \mathbf{W}_{re}^{(1)} \mathbf{u}_i^{re} \parallel \mathbf{W}_{re}^{(1,2)} \mathbf{v}_j^{re} \right]\right),$$

where  $\mathbf{W}_{re}^{(1,2)}$  is the weight matrix for every anchor node in  $G^{(2)}$  as the recipient and  $\mathbf{a}_{re}^{(1,2)T}$  is also a weight vector.

Inter-network anchor attention mechanism can make a great contribution to effective cross-network information transfer. It handles the problem of network characteristic differences by making our model focus on more important networks with useful information.

Besides, to make multiple coefficients easily comparable across different nodes, we normalize all the coefficients mentioned above across all choices of intra-network neighbors and inter-network

partners using the softmax function. Thus, the four kinds of attention coefficients of the concerned nodes within networks and across networks can be rewritten into a unified formula:

$$\begin{aligned}\alpha^{op}(u_i, u_j/v_j) &= \text{softmax}_{u_j/v_j}(e^{op}(u_i, u_j/v_j)) \\ &= \exp(e^{op}(u_i, u_j/v_j)) / \sum_{u_k \in \mathcal{N}^{op}(u_i)} \exp(e^{op}(u_i, u_k))\end{aligned}$$

, where  $op \in \{in, re\}$  indicates the role of the target node. And the denominator is set as

$$\sum_{u_k \in \mathcal{N}^{op}(u_i)} \exp(e^{op}(u_i, u_k)) + \sum_{v_k \in \mathcal{N}^a(u_i)} \exp(e^{op}(u_i, v_k))$$

### 3.2 Multiple Aligned Network Embedding

With the hierarchical graph attention introduced in the previous section, we will introduce the cross-network embedding framework HGANE in this part. HGANE is based on the cross-network graph neural network model, which extends the traditional graph neural network (GNN) model [16] to the multiple aligned social networks scenario. According to the principle of GNN, embeddings can capture the localized structural features by utilizing information propagated from the intra-network social neighbors. What's more, it can preserve more comprehensive features by leveraging cross-network information transferred by anchor links. Therefore, HGANE learns the node representations by aggregating information from both the intra-network neighbors and inter-network partners. At the same time, HGANE takes advantage of the hierarchical graph attention mechanism to focus on more important information to handle the problems of network characteristic differences and link directivity challenges. The architecture of HGANE is illustrated in Figure 3.

Each node is represented by two embeddings,  $\mathbf{u}_i^{in}$  and  $\mathbf{u}_i^{re}$ , according to its two role in social networks. The implicit initiator and recipient representations of the node are represented as  $\mathbf{u}_i^{in'} \in \mathbb{R}^{d'}$  and  $\mathbf{u}_i^{re'} \in \mathbb{R}^{d'}$  of the potentially different dimension  $d'$ .

The initiator embedding of  $u_i$ ,  $\mathbf{u}_i^{in}$ , which indicates its features as the initiator in the social network, depends on its intra-network recipient neighbors. Therefore, for the node  $u_j \in \mathcal{N}^i(u_i)$ ,  $\mathbf{u}_j^{re}$  can contribute to  $\mathbf{u}_i^{in}$  with the coefficient,  $\alpha^{in}(u_i, u_j)$ , determined by intra-network initiator attention. We define the *intra-network neighbor recipient contribution* (NRC) from  $u_j$  to  $u_i$  as

$$NRC(u_i, u_j) = \alpha^{in}(u_i, u_j) \mathbf{W}_{re}^{(1)} \mathbf{u}_j^{re}$$

As to inter-network anchor partners, node embeddings can preserve more comprehensive information by taking their features of the same role in other networks into consideration. For the anchor node  $v_j \in \mathcal{N}^a(u_i)$ , its initiator embedding  $\mathbf{v}_j^{in}$  also contribute to the initiator embedding  $\mathbf{u}_i^{in}$  of the target node. To overcome the problem of network characteristic differences, the inter-network initiator attention will compute the weights of information from different networks. The *inter-network partner initiator contribution* (PIC) from  $v_j$  to  $u_i$  is introduced as

$$PIC(u_i, v_j) = \alpha^{in}(u_i, v_j) \mathbf{W}_{in}^{(1,2)} \mathbf{v}_j^{in}$$

Formally, we can obtain the initiator embedding of  $u_i$  by aggregating the intra-network neighbor recipient contribution and the inter-network partner initiator contribution as

$$\mathbf{u}_i^{in'} = \sigma \left( \sum_{u_j \in \mathcal{N}^i(u_i)} NRC(u_i, u_j) + \sum_{v_j \in \mathcal{N}^a(u_i)} PIC(u_i, v_j) \right),$$

where  $\sigma$  denotes a nonlinearity. Besides, the recipient embedding of  $u_i$ ,  $\mathbf{u}_i^{re}$ , can be generated in the similar way. Inter-network and inter-network recipient attention will determine the importance of different related nodes' contribution. As the recipient,  $u_i$  is naturally characterized by its intra-network initiator neighbors who actively follow it in social networks. Such intuition leads to the contribution from every node, such as  $u_j \in \mathcal{N}^r(u_i)$ , to  $u_i$ . The *intra-network neighbor initiator contribution* (NIC) from  $u_j$  to  $u_i$  can be defined as

$$NIC(u_i, u_j) = \alpha^{re}(u_i, u_j) \mathbf{W}_{in}^{(1)} \mathbf{u}_j^{in}$$

The inter-network anchor partners provide more information about the recipient role of the target node from different sources. Thus, the recipient embedding of  $u_i$  will aggregate the information from other networks by the anchor nodes with different weights computed by the inter-network recipient attention in other networks. We can obtain the *inter-network partner recipient contribution* (PRC) from  $u_j$  to  $u_i$  to be

$$PRC(u_i, v_j) = \alpha^{re}(u_i, v_j) \mathbf{W}_{re}^{(1,2)} \mathbf{v}_j^{re}$$

By combining these contributions from intra-network initiator neighbors and inter-network partners, the recipient embedding of  $u_i$  can be represented formally as

$$\mathbf{u}_i^{re'} = \sigma \left( \sum_{u_j \in \mathcal{N}^r(u_i)} NIC(u_i, u_j) + \sum_{v_j \in \mathcal{N}^a(u_i)} PRC(u_i, v_j) \right)$$

To stabilize the learning process of node embeddings, we have found extending our mechanism to employ multi-head attention to be beneficial, inspired by Vaswani et al. 2017 [20]. Specifically,  $K$  independent attention mechanisms execute the above transformation, and then their embeddings are concatenated, resulting in the following initiator and recipient feature representations:

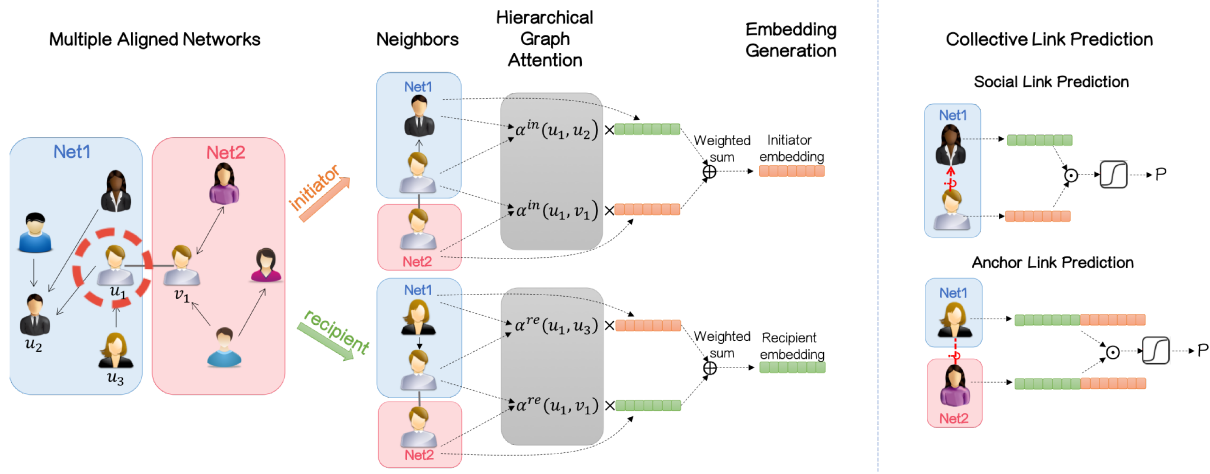
$$\begin{aligned}\mathbf{u}_i^{in'} &= \parallel_{k=1}^K \sigma \left( \sum_{u_j \in \mathcal{N}^i(u_i)} NRC_k(u_i, u_j) + \sum_{v_j \in \mathcal{N}^a(u_i)} PIC_k(u_i, v_j) \right) \\ \mathbf{u}_i^{re'} &= \parallel_{k=1}^K \sigma \left( \sum_{u_k \in \mathcal{N}^r(u_i)} NIC_k(u_i, u_j) + \sum_{v_j \in \mathcal{N}^a(u_i)} PRC_k(u_i, v_j) \right)\end{aligned}$$

where NRC, PIC, NIC and PRC with the subscript  $k$  in the formulas denote the contributions computed with the  $k$ -th hierarchical attention mechanism.

With reference to above equations, the formula derivation of computing the node embeddings for  $G^{(2)}$  can be obtained in the similar way. They are not listed due to the page limit.

### 3.3 Collective Link Prediction Oriented Network Embedding Framework

The embeddings of each node in multiple aligned networks can be generated by aggregating information from both the intra-network neighbors and inter-network partners as introduced in last section. In this part, we will introduce the network embedding optimization framework oriented to collective link prediction. The task includes



**Figure 3: HGANE model architecture.** Each node is represented by the initiator and recipient embeddings (the orange and green little squares) in the aligned networks. The left plot provides the example of learning node embeddings ( $u_1$  in Net1) by aggregating information from both the intra-network neighbors ( $u_2, u_3$  in Net1) and inter-network partners ( $v_1$  in Net2), which is weighted by hierarchical graph attention. The right plot illustrates how HGANE resolves the collective link prediction with the learned embeddings.

the simultaneous prediction of the social links within each network and the anchor links between every two networks.

For a node pair  $(u_i, u_j)$  within the social network, we define the probability of the intra-network social link formation from the initiator  $u_i$  pointing to the recipient  $u_j$  as

$$p(u_i, u_j) = \sigma(\mathbf{u}_i^{inT} \cdot \mathbf{u}_j^{re}),$$

where  $\sigma(x) = 1/(1+\exp(-x))$  is the sigmoid function. And we adopt the approach of negative sampling [13] to define the objective of intra-network social link formation from the initiator  $u_j$  to the recipient  $u_i$  as

$$\mathcal{L}_{soc}(u_i, u_j) = \log p(u_i, u_j) + \sum_{\{(u_m, u_n)\}} \log(1 - p(u_m, u_n)),$$

where  $\{(u_m, u_n)\}$  denotes the set of the negative social links random sampled from the unknown links among nodes in  $G^{(1)}$ . In the objective, the first term models the existing social links while the second term models the negative links. By adding the objective of each intra-network social link, the final objective for  $G^{(1)}$  can be formally represented as

$$\mathcal{L}^{(1)} = \sum_{(u_i, u_j) \in E^{(1)}} \mathcal{L}_{soc}(u_i, u_j)$$

Similarly, we can define the objective for the embedding results for  $G^{(2)}$ , which can be formally represented as  $\mathcal{L}^{(2)}$ .

Besides, anchor nodes reflect information of same users. Therefore, their features tend to be in a close region in the embedding space whether as the initiator or the initiator. For the cross-network node pair  $(u_i, v_j)$  where  $u_i \in E^{(1)}$  and  $v_j \in E^{(2)}$ , we concatenate the initiator and recipient embeddings of each node to define the probability of the inter-network anchor link formation as

$$p(u_i, v_j) = \sigma((\mathbf{u}_i^{in} \parallel \mathbf{u}_i^{re})^T \cdot (\mathbf{v}_j^{in} \parallel \mathbf{v}_j^{re}))$$

Similarly with the objective of the intra-network social link formation, the objective of node alignment with negative sampling can be defined as

$$\mathcal{L}_{ach}(u_i, v_j) = \log p(u_i, v_j) + \sum_{\{(u_m, v_n)\}} \log(1 - p(u_m, v_n)),$$

where  $\{(u_m, v_n)\}$  denotes the set of the negative anchor links random sampled from the unknown anchor links across  $G^{(1)}$  and  $G^{(2)}$ . By aligning anchor nodes, we can leverage information from multiple sources to learn the node embeddings comprehensively. Information transfer across networks is achieved based on every inter-network anchor link. Formally, the information transfer objective between  $G^{(1)}$  and  $G^{(2)}$  is represented by summing up the the objective of each anchor link as

$$\mathcal{L}^{(1,2)} = \sum_{(u_i, v_j) \in A^{(1,2)}} (\mathcal{L}_{ach}(u_i, v_j))$$

To incorporate the collective link prediction task into a unified framework, we learn the node representations with rich information by jointly training the objective function including the objective for networks  $G^{(1)}$ ,  $G^{(2)}$ , and the objective of information transfer, which can be denoted as

$$\mathcal{L}(G^{(1)}, G^{(2)}) = \mathcal{L}^{(1)} + \mathcal{L}^{(2)} + \alpha \cdot \mathcal{L}^{(1,2)} + \beta \cdot \mathcal{L}_{reg}$$

The parameter  $\alpha$  denotes the weight of the information transfer objective to balance between the several prediction tasks of both the intra-network social link and the inter-network anchor link. In the objective function, the term  $\mathcal{L}_{reg}$  is added to avoid overfitting and the parameter  $\beta$  denotes the weight of it. By optimizing the above objective function, the node embeddings can be learned to resolve the collective link prediction problem effectively.

**Table 2: Statistics of datasets**

| Dataset    | #Nodes | #Social Links | #Anchor Links |
|------------|--------|---------------|---------------|
| Twitter    | 5,223  | 164,920       | 3,388         |
| Foursquare | 5,392  | 76,972        |               |
| Facebook   | 4,137  | 57,528        | 4,137         |
| Twitter    | 4,137  | 147,726       |               |

## 4 EXPERIMENT

### 4.1 Datasets

We conducted experiments using two real-world aligned social networks: Twitter-Foursquare and Facebook-Twitter(Statistical information in Table 2):

- **Twitter-Foursquare** [9]: Twitter is the most popular world-wide microblog site and Foursquare is the famous location-based social network. There are 5,223 users and 164,920 follow links in Twitter and 5,392 users and 76,972 social links in Foursquare. Among these crawled Foursquare users, 3,388 of them are aligned by anchor links with Twitter.
- **Facebook-Twitter** [2]: Facebook is another worldwide online social media. The Facebook and Twitter accounts of 4,137 users were crawled. Every node has a counterpart in the other network. There are 57,528 social links in Facebook and 147,726 follow links in Twitter among these users.

It is noted that these two datasets were crawled respectively and there is no overlap of these two Twitter subnetworks.

**Source Code:** The source code of HGANE is available in <http://github.com/yzjiao/HierarchicalGraphAttention>.

### 4.2 Comparison Methods

**Table 3: Comparison of different models**

| Method   | Multiple Networks | Links Directi. Differe. | Network Charact. Differe. | Predict Social Link | Predict Anchor Link |
|----------|-------------------|-------------------------|---------------------------|---------------------|---------------------|
| DeepWalk |                   |                         |                           | ✓                   |                     |
| Node2Vec |                   |                         |                           | ✓                   |                     |
| GAT      |                   |                         |                           | ✓                   |                     |
| IONE     | ✓                 | ✓                       |                           |                     | ✓                   |
| DIME     | ✓                 | ✓                       |                           | ✓                   |                     |
| MNN      | ✓                 | ✓                       |                           | ✓                   |                     |
| CLF      | ✓                 |                         |                           | ✓                   | ✓                   |
| HGANE    | ✓                 | ✓                       | ✓                         | ✓                   | ✓                   |

The network embedding methods used in the experiment are listed as follows (summarized in Table 3):

- **DeepWalk** [15]: Skip-gram based vertex embedding method for a single network that extends the word2vec [14] to the network scenario.
- **Node2Vec** [6]: Word-to-vector approach for a single network that modifies the random walk strategy in DeepWalk into a more sophisticated schemes.

- **GAT** [20]: A neural network architecture for a single network to learn node representation by leveraging masked self-attention layers.
- **IONE** [12]: A representation learning model for multiple aligned network by preserving the proximity of users with “similar” followers/followees in the embedded space for network alignment.
- **DIME** [30]: An embedding framework for multiple heterogeneous aligned network with aligned autoencoders to transfer information and improve the link prediction in emerging networks.
- **MNN** [1]: A multi-neural-network framework for intra-network link prediction over aligned networks. It is not suitable for anchor link prediction as it assigns anchor users with the same feature vectors.
- **CLF** [23]: A method aiming at collective link prediction by propagating the probabilities of predicted links across the partially aligned networks with collective random walk.

### 4.3 Experiment Setting

In our experiment, we will target on the collective link prediction task and concern the performance of social link prediction in  $G^{(1)}$ ,  $G^{(2)}$ , and anchor link prediction across these two networks. These three subtasks will be denoted as Soc1, Soc2, Ach in the experiment results. As link prediction is regarded as a binary classification task, the performance will be evaluated with Area Under the Curve (AUC) metric.

All the existing links in the two aligned networks are used as the positive link set, including social links within two networks and anchor links across these two networks. We sample a subset of unknown links among nodes in the same network randomly as the negative social link set, which is of the double size of the positive social link set. The negative anchor link set is generated by the random sample of unknown cross-network links. The size of the negative anchor link set is five times of that of the positive set. A proportion of the links in the positive and negative sets are sampled as the training set, the rest as the test set.

For our embedding framework HGANE, we initialize the initiator and recipient features of each node with the common initiator and recipient neighbor features within its networks. There are two attention-based layers involved for each network. The first layer consists of  $K = 8$  attention heads computing 256 features each, followed by an exponential linear unit (ELU) [4] nonlinearity. The second layer is a single attention head to compute node embeddings, followed by a softmax activation. The dimension of the embeddings is 100. During training, we apply dropout [17] to the normalized attention coefficients. And we train for 3000 epochs using the Adam optimizer [8] with the learning rate of 0.005. The parameters  $\alpha = 1.0$  and  $\beta = 0.0005$  are used in the experiments.

For the comparison methods that target at one single network, such as Node2vec, DeepWalk and GAT, we preprocess the datasets by merging two networks into one and regarding anchor links as social links within networks. We apply the linear SVM classifier for those embedding methods that can’t directly predict the formation

**Table 4: Performance comparison with different methods. Soc1, Soc2 and Ach indicate social link prediction in the first and second network and anchor link prediction respectively.**

| Dataset              | Method   | Training Ratio |              |              |              |              |              |              |              |              |              |              |              |
|----------------------|----------|----------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
|                      |          | 0.2            |              |              | 0.4          |              |              | 0.6          |              |              | 0.8          |              |              |
|                      |          | Soc1           | Soc2         | Ach          | Soc1         | Soc2         | Ach          | Soc1         | Soc2         | Ach          | Soc1         | Soc2         | Ach          |
| Twitter & Foursquare | DeepWalk | 75.8%          | 72.5%        | 57.9%        | 80.3%        | 76.9%        | 63.1%        | 82.2%        | 79.7%        | 67.3%        | 85.7%        | 82.5%        | 75.4%        |
|                      | Node2Vec | 82.5%          | 77.4%        | 64.3%        | 84.6%        | 80.9%        | 66.1%        | 86.4%        | 84.3%        | 72.1%        | 89.3%        | 88.3%        | 78.9%        |
|                      | GAT      | 85.5%          | 78.2%        | 65.5%        | 91.5%        | 86.9%        | 68.9%        | 92.5%        | 90.3%        | 75.8%        | 92.6%        | 92.3%        | 80.9%        |
|                      | IONE     | 83.2%          | 75.7%        | 72.1%        | 86.2%        | 81.7%        | 78.0%        | 88.2%        | 84.7%        | 85.6%        | 88.7%        | 84.7%        | 87.4%        |
|                      | DIME     | 85.1%          | 76.2%        | 74.8%        | 88.4%        | 80.3%        | 76.3%        | 89.8%        | 83.0%        | 82.6%        | 92.0%        | 85.2%        | 84.9%        |
|                      | MNN      | 89.2%          | 72.4%        | -            | 92.9%        | 81.1%        | -            | 94.8%        | 86.1%        | -            | 96.3%        | 87.6%        | -            |
|                      | CLF      | 84.5%          | 78.7%        | 70.9%        | 86.7%        | 80.5%        | 75.2%        | 90.9%        | 84.2%        | 83.1%        | 92.3%        | 86.5%        | 87.1%        |
|                      | HGANE    | <b>94.4%</b>   | <b>90.3%</b> | <b>76.7%</b> | <b>96.4%</b> | <b>95.1%</b> | <b>85.8%</b> | <b>97.1%</b> | <b>96.8%</b> | <b>90.0%</b> | <b>97.5%</b> | <b>97.5%</b> | <b>93.0%</b> |
| Facebook & Twitter   | DeepWalk | 76.3%          | 70.3%        | 55.8%        | 81.5%        | 75.2%        | 70.9%        | 84.0%        | 81.6%        | 77.7%        | 90.9%        | 86.5%        | 78.9%        |
|                      | Node2Vec | 83.0%          | 81.5%        | 58.6%        | 86.6%        | 85.7%        | 76.2%        | 88.8%        | 87.5%        | 81.0%        | 91.3%        | 88.2%        | 83.2%        |
|                      | GAT      | 87.3%          | 86.1%        | 60.2%        | 92.0%        | 90.0%        | 78.5%        | 94.7%        | 92.8%        | 83.5%        | 95.7%        | 93.4%        | 85.5%        |
|                      | IONE     | 82.8%          | 79.1%        | 77.9%        | 85.9%        | 82.6%        | 85.4%        | 87.4%        | 85.1%        | 89.4%        | 90.9%        | 89.1%        | 92.1%        |
|                      | DIME     | 87.1%          | 86.2%        | 74.3%        | 88.4%        | 87.3%        | 81.9%        | 89.8%        | 90.0%        | 85.1%        | 94.0%        | 92.2%        | 87.5%        |
|                      | MNN      | 88.6%          | 87.1%        | -            | 92.4%        | 91.3%        | -            | 94.4%        | 93.1%        | -            | 95.7%        | 94.8%        | -            |
|                      | CLF      | 84.9%          | 81.1%        | 80.5%        | 88.7%        | 85.9%        | 84.2%        | 91.4%        | 88.9%        | 87.6%        | 93.1%        | 90.2%        | 90.4%        |
|                      | HGANE    | <b>91.8%</b>   | <b>90.9%</b> | <b>84.8%</b> | <b>95.2%</b> | <b>94.8%</b> | <b>93.4%</b> | <b>97.1%</b> | <b>96.9%</b> | <b>95.8%</b> | <b>98.1%</b> | <b>97.5%</b> | <b>97.1%</b> |

**Table 5: Validation of the design of represent each node with two embeddings to resolve the link directivity differences problem. Our full model outperforms two variants with either the initiator or recipient features.**

| Feature   | Twitter&Foursquare |              |              | Facebook&Twitter |              |              |
|-----------|--------------------|--------------|--------------|------------------|--------------|--------------|
|           | Soc1               | Soc2         | Ach          | Soc1             | Soc2         | Ach          |
| initiator | 93.4%              | 93.2%        | 85.2%        | 97.0%            | 94.9%        | 95.8%        |
| recipient | 93.0%              | 93.7%        | 85.6%        | 97.1%            | 95.1%        | 96.2%        |
| both      | <b>97.2%</b>       | <b>96.8%</b> | <b>93.0%</b> | <b>98.1%</b>     | <b>97.5%</b> | <b>97.1%</b> |

of links. Notably, we have chosen optimal hyper-parameters carefully for different baselines in this paper to ensure the fairness of comparison experiments.

#### 4.4 Experiment Result

In the collective link prediction task, we compare the performance of eight different embedding methods under different training rate  $\lambda \in \{20\%, 40\%, 60\%, 80\%\}$ . Table 4 shows the performance of our model and other seven baseline methods evaluated by AUC with different training rate  $\lambda$ . The method we proposed in this paper, HGANE, performs much better than the other methods in the three subtasks simultaneously, which shows its effectiveness in the collective link prediction task. HGANE incorporates the task-oriented objectives into consideration and thus balance between the prediction tasks of both the intra-network social link and inter-network anchor link simultaneously.

Considering the experiments with different training rate  $\lambda$ , as the ratio drops, the performance of all the methods deteriorates. However, the performance degradation of the proposed model is

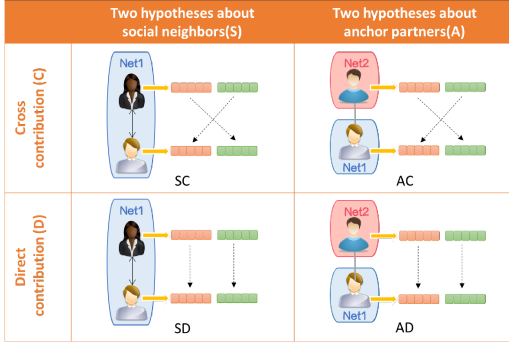
rather moderate compared to other methods since we leverage the information of the multiple aligned networks and handle the problem of network characteristic differences. Even when the training rate  $\lambda$  is as low as 20%, the baseline models will suffer from the information sparsity a lot, but our model can still obtain very good performance.

To demonstrate the effectiveness of considering the directivity, we compare our full model to its two variants with either the initiator features or the recipient features. The experiment results show in Table 5 with the training ratio as 0.8. We found that the two variants can give better results than other baselines but their performance is much inferior to that of the full model. According to the experimental statistics of two datasets, the performance of social link prediction within the more dense network can be improved more by distinguishing nodes' initiator and recipient roles.

#### 4.5 Hypothesis Verification

As mentioned in the method part, in our framework, each node is represented with the initiator and recipient features and it is crucial to determine how to aggregate information from the neighbors connected by social links and anchor links. If the neighbor's initiator and recipient features contribute to the target node's recipient and initiator respectively, we name it as cross-contribution. Conversely, if the initiator and recipient features of two neighbor nodes are related correspondingly, we name it direct-contribution. By combining two mechanisms with two kinds of links, there are four different hypotheses as illustrated in Figure 4.

The hypotheses SC+AD is adopted in our framework. To validate it, we study the variants of our full model with the other three hypotheses and compared their performances on the collective link prediction task in Table 6. The experimental results indicate the



**Figure 4: Four hypotheses about how the neighbors contribute to the target node. There are two kinds of contribution modes for social neighbors and anchor partners respectively.**

model with SC+AD can achieve the best performance in both social link prediction and anchor link prediction. If direct-contribution is replaced with cross-contribution for anchor links, AUC of the anchor link prediction decreases a lot in two datasets. And the performances of the social link prediction are affected if we adopt direct-contribution for social links.

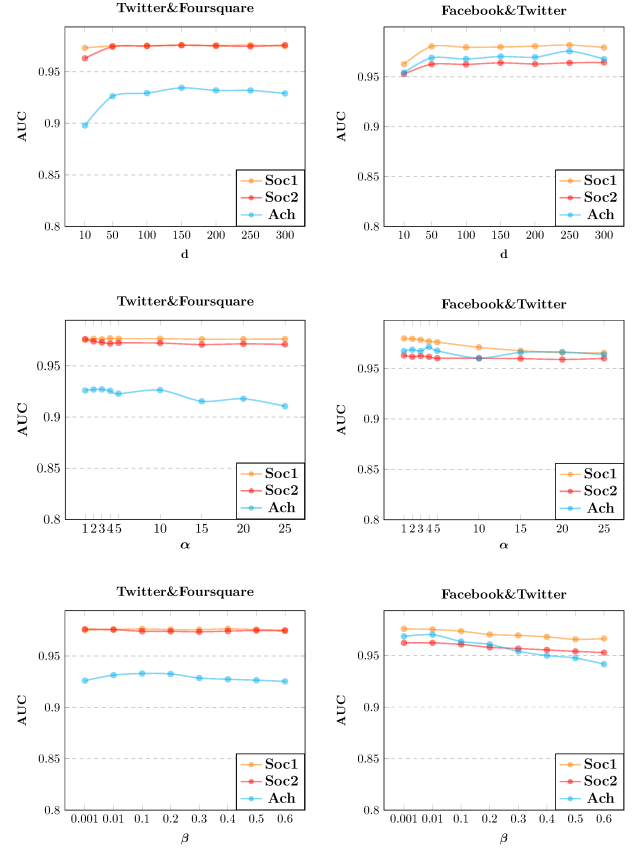
**Table 6: Hypothesis verification. SC+AD is adopted in our framework and achieves the best results.**

| Hypothesis   | Twitter&Foursquare |              |              | Facebook&Twitter |              |              |
|--------------|--------------------|--------------|--------------|------------------|--------------|--------------|
|              | Soc1               | Soc2         | Ach          | Soc1             | Soc2         | Ach          |
| SC+AC        | 97.2%              | 96.0%        | 89.4%        | 98.1%            | 97.0%        | 96.0%        |
| SD+AD        | 91.8%              | 91.3%        | 82.8%        | 97.0%            | 95.2%        | 95.3%        |
| SD+AC        | 91.8%              | 91.3%        | 82.1%        | 96.9%            | 95.3%        | 94.3%        |
| <b>SC+AD</b> | <b>97.2%</b>       | <b>96.7%</b> | <b>92.4%</b> | <b>98.1%</b>     | <b>97.1%</b> | <b>96.7%</b> |

#### 4.6 Parameter Analysis

Now we examine the influence of three key parameters in our framework: the embedding size  $d$  and the weight of the information transfer objective  $\alpha$  and the weight of the regularization  $\beta$ . The three subfigures on the left in Figure 5 show the sensitivity analysis on the first dataset while the rest is about the second dataset. The results in the first two subfigures indicate that setting the embedding size  $d$  to 100 can provide the best performance on both two datasets. Even when  $d$  is as low as 10, our model can achieve good results on three kinds of link prediction simultaneously.

The parameter  $\alpha$  denotes the strength of aligning the two networks. The two subfigures about  $\alpha$  in the middle show how different values of  $\alpha$  can affect the results on different datasets. The optimal  $\alpha$  is near 1.0. When setting  $\alpha$  in  $[1, 3]$ , all the link prediction tasks perform well and stably. Anchor link prediction and social link prediction in sparser networks are affected as  $\alpha$  increases. However, social link prediction in the dense networks is still stable. For the weight parameter  $\beta$ , the best setting is in  $[0.1, 0.2]$  according to the last two subfigures. It has a certain impact on anchor link prediction



**Figure 5: Hyperparameter analysis. Our method is robust to choices of  $d$ ,  $\alpha$  and  $\beta$  on two datasets**

across the network while social link prediction within the network is not that sensitive to the parameter  $\beta$  on both two datasets.

## 5 RELATED WORK

**Multi-Network Analysis** Traditional network embedding methods focus on one single network [6, 15, 32] and suffer from the data insufficiency problem in the cold start scenarios. Therefore, multi-network analysis has been a hot research topic and studied for data enrichment for several years on which dozens of works have been published [24, 28]. Some work studied on information transfer across networks by anchor links to improving the quality of inter-network link prediction [1, 25, 26, 30, 31]. Besides, many existing works aim at anchor link formation prediction automatically [9, 27]. However, most of these works study either intra-network or inter-network link prediction tasks separately. Zhang et al. first proposed the collective link prediction task [23]. The existing methods mostly ignore the contradiction of different characteristics of aligned networks or adopt fixed parameters to control the proportion of information diffused across networks, which usually need to be fine-tuned manually. Besides, these works also fail to consider the link connectivity of the links within and across networks.

**Neural attention mechanism** Neural attention mechanism has inspired many state-of-the-art models in several machine learning tasks including image caption generation [22], machine translation [5, 19] and semantic role labeling [18]. Its effectiveness is owed to making the model focus on more important detailed information and neglecting the useless information. In recent years, some works have also investigated the use of attention on graphs [3, 20, 29]. Our work propose the hierarchical graph attention to model multiple aligned networks and overcome network characteristics contradiction to transfer more effective information across networks.

## 6 CONCLUSION

In this paper, we study the collective link prediction problem in multiple aligned social networks. We propose a novel application oriented network embedding framework, namely Hierarchical Graph Attention based Network Embedding (HGANE) to learn node embeddings. The hierarchical graph attention mechanism is introduced to resolve the network characteristic differences and link directivity differences. We conduct detailed empirical evaluations using several real-world datasets and the results demonstrate that our model outperforms other competitive approaches and handles the collective link prediction problems effectively.

## ACKNOWLEDGMENTS

This work is supported in part by the National Natural Science Foundation of China Projects No.U1636207 and NSF through grant IIS-1763365. This work is also partially supported by the Shanghai Science and Technology Development Fund No. 16JC1400801 and the National Natural Science Foundation of China Projects No.91546105.

## REFERENCES

- [1] Xuezhi Cao, Haokun Chen, Xuejian Wang, Weinan Zhang, and Yong Yu. 2018. Neural link prediction over aligned networks. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- [2] Xuezhi Cao and Yong Yu. 2016. ASNets: A benchmark dataset of aligned social networks for cross-platform user modeling. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management*. ACM, 1881–1884.
- [3] Edward Choi, Mohammad Taha Bahadori, Le Song, Walter F Stewart, and Jimeng Sun. 2017. GRAM: graph-based attention model for healthcare representation learning. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 787–795.
- [4] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. 2015. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). *CoRR* abs/1511.07289 (2015). [arXiv:1511.07289](http://arxiv.org/abs/1511.07289) <http://arxiv.org/abs/1511.07289>
- [5] Jonas Gehring, Michael Auli, David Grangier, and Yann Dauphin. 2017. A Convolutional Encoder Model for Neural Machine Translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vol. 1. 123–135.
- [6] Aditya Grover and Jure Leskovec. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 855–864.
- [7] Adrien Guille, Hakim Hacid, Cecile Favre, and Djamel A Zighed. 2013. Information diffusion in online social networks: A survey. *ACM Sigmod Record* 42, 2 (2013), 17–28.
- [8] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).
- [9] Xiangnan Kong, Jiawei Zhang, and Philip S Yu. 2013. Inferring anchor links across multiple heterogeneous social networks. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. ACM, 179–188.
- [10] Vincent Leroy, B Barla Cambazoglu, and Francesco Bonchi. 2010. Cold start link prediction. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 393–402.
- [11] Yongjun Li, You Peng, Zhen Zhang, Hongzhi Yin, and Qunqun Xu. 2018. Matching user accounts across social networks based on username and display name. *World Wide Web* (2018), 1–23.
- [12] Li Liu, William K Cheung, Xin Li, and Lejian Liao. 2016. Aligning Users across Social Networks Using Network Embedding.. In *IJCAI*. 1774–1780.
- [13] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013). [arXiv:1301.3781](http://arxiv.org/abs/1301.3781) <http://arxiv.org/abs/1301.3781>
- [14] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.
- [15] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 701–710.
- [16] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2009. The graph neural network model. *IEEE Transactions on Neural Networks* 20, 1 (2009), 61–80.
- [17] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.
- [18] Zhixing Tan, Mingxuan Wang, Jun Xie, Yidong Chen, and Xiaodong Shi. 2018. Deep semantic role labeling with self-attention. In *Thirty-Second AAAI Conference on Artificial Intelligence*.
- [19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*. 5998–6008.
- [20] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2017. Graph Attention Networks. *CoRR* abs/1710.10903 (2017). [arXiv:1710.10903](http://arxiv.org/abs/1710.10903) <http://arxiv.org/abs/1710.10903>
- [21] Yaqing Wang, Chunyan Feng, Ling Chen, Hongzhi Yin, Caili Guo, and Yunfei Chu. 2018. User identity linkage across social networks via linked heterogeneous network embedding. *World Wide Web* (2018), 1–22.
- [22] Baj XuK, CourvilleA KirosR, et al. 2015. Show, attend and tell: Neural image caption generation with visual attention. *Proceedings of the International Conference on Machine Learning*. Lille, France 2048 (2015), 2057.
- [23] Qianyi Zhan, Jiawei Zhang, and S Yu Philip. 2018. Integrated anchor and social link predictions across multiple social networks. *Knowledge and Information Systems* (2018), 1–24.
- [24] Jiawei Zhang. 2018. Social Network Fusion and Mining: A Survey. *arXiv preprint arXiv:1804.09874* (2018).
- [25] Jiawei Zhang, Xiangnan Kong, and S Yu Philip. 2013. Predicting social links for new users across aligned heterogeneous social networks. In *2013 IEEE 13th International Conference on Data Mining*. IEEE, 1289–1294.
- [26] Jiawei Zhang, Xiangnan Kong, and Philip S Yu. 2014. Transferring heterogeneous links across location-based social networks. In *Proceedings of the 7th ACM international conference on Web search and data mining*. ACM, 303–312.
- [27] Jiawei Zhang and S Yu Philip. 2015. Multiple anonymized social networks alignment. In *2015 IEEE International Conference on Data Mining*. IEEE, 599–608.
- [28] Jiawei Zhang and S Yu Philip. 2019. *Broad Learning Through Fusions: An Application on Social Networks*. Springer.
- [29] Jiani Zhang, Xingjian Shi, Junyuan Xie, Hao Ma, Irwin King, and Dit-Yan Yeung. 2018. Gaan: Gated attention networks for learning on large and spatiotemporal graphs. *arXiv preprint arXiv:1803.07294* (2018).
- [30] Jiawei Zhang, Congying Xia, Chenwei Zhang, Limeng Cui, Yanjie Fu, and S Yu Philip. 2017. BL-MNE: emerging heterogeneous social network embedding through broad learning with aligned autoencoder. In *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE, 605–614.
- [31] Jiawei Zhang, Philip S Yu, and Zhi-Hua Zhou. 2014. Meta-path based multi-network collective link prediction. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1286–1295.
- [32] Yao Zhang, Yun Xiong, Xiangnan Kong, and Yangyong Zhu. 2017. Learning node embeddings in interaction graphs. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 397–406.
- [33] Xiaoping Zhou, Xun Liang, Xiaoyong Du, and Jichao Zhao. 2018. Structure based user identification across social networks. *IEEE Transactions on Knowledge and Data Engineering* 30, 6 (2018), 1178–1191.