Who Proposed the Relationship? — Recovering the Hidden Directions of Undirected Social Networks[*]

Jun Zhang[1,2,3,4] Chaokun Wang[2,3,4] Jianmin Wang[2,3,4]
[1] Department of Computer Science and Technology, Tsinghua University
[2] School of Software, Tsinghua University
[3] Tsinghua National Laboratory for Information Science and Technology
[4] Key Laboratory for Information System Security, Ministry of Education, P. R. China
zhang-jun10@mails.tsinghua.edu.cn, chaokun@tsinghua.edu.cn, jimwang@tsinghua.edu.cn

ABSTRACT

Together with the sign (positive or negative) and strength (strong or weak), the directionality is also an important property of social ties, though usually ignored in undirected social networks for its invisibility. However, we believe most social ties are natively directed, and the awareness of directionality can improve our understanding about the network structures and further benefit social network analysis and mining tasks. Thus it's appealing to study whether there exist interesting patterns about directionality in social networks and whether we can learn the directions for undirected networks based on these patterns.

In this study, we engage in the investigation of directionality patterns on real-world directed social networks and summarize our findings using four consistency hypotheses. Based on these hypotheses, we propose *ReDirect*, an optimization framework which makes it possible to infer the hidden directions of undirected social ties based on the network topology only. This general framework can incorporate various predictive models under specific scenarios. Furthermore, we show how to improve ReDirect by introducing semi/self-supervision in the framework and how to construct the self-labeled training data using simple but effective heuristics. Experimental results show that even without external information, our approach can recover the directions of networks effectively.

Moreover, we're quite surprising to find that ReDirect can benefit predictive tasks remarkably, with a case study of link prediction. In experiments the redirected networks inferred using ReDirect are proven much more informative than original undirected ones and can improve the prediction performance significantly. It convinces us that ReDirect can be a beneficial general data preprocess tool for various network analysis and mining tasks by uncovering the hidden directions of undirected social networks.

Categories and Subject Descriptors

H.2.8 [**DATABASE MANAGEMENT**]: Database Applications—*Data mining*; J.4 [**Computer Applications**]: Social and Behavior Sciences

Keywords

ReDirect; Tie Direction Inference; Directionality; Social Networks

1. INTRODUCTION

The world is not flat. In a romantic relationship, usually, a boy firstly confesses his love for the girl he likes, and when the girl agrees, they become a couple. In an academic collaboration relationship, a junior researcher often asks a senior one to co-author papers and this relationship is built when the latter agrees. In Facebook or LinkedIn, the creation of a relationship usually involves the process that one initializes a friend request to another who would respond to that later. Many social ties are born with directionality; however, the directions are usually hidden and unobservable in such networks which are shown undirected, such as couple relationships, academic social networks, Facebook and LinkdeIn. Thus an interesting question is: Can we infer the hidden directions of ties in undirected social networks?

Though usually invisible, the directionality, together with the sign (positive or negative) [17] and strength (strong or weak) [6, 35], is also one of the rich inherent properties of social ties. It explains *the creation of social ties*, i.e. who proposed the relationship to the other in a social tie? We say the *proposer* is the one who proposes the relationship firstly, and the *responder* the one who responds to that later. In the above examples, the boy in the romantic relationship, the junior researcher in the collaboration relationship and the friend-request initiator in Facebook or LinkedIn are the proposers, while the others who respond to the relationship proposals are the responders. Thus, the hidden direction of the observable undirected social tie is assumed as "*proposer* → *responder*". We also call the responder as a *followee* of the proposer, and the proposer as a *follower* of the responder.

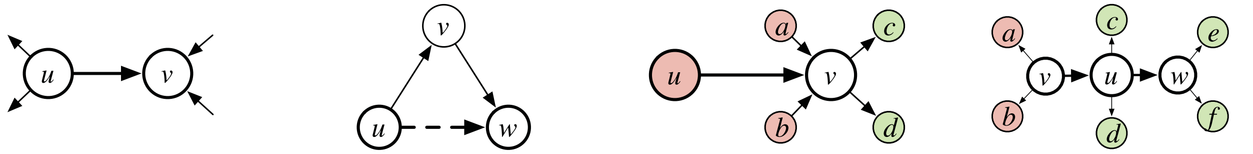
The awareness of directionality can benefit various social network analysis and mining tasks. Social ties have been proven beneficial in many tasks, such as link prediction [21], rating prediction [31], product recommendation [24] and community discovery [29]; however, they were usually regarded as plain binary relational ties (e.g., friends or not). The ignorance of directionality may introduce noise and lead to inaccurate models because one's followers and followees may have different influence on one's behaviors. This assumption is verified in our experimental study in Sec. 6.2, and we're excited to find that the direction awareness would increase the performance of predictive tasks on social networks significantly, especially link prediction.

1.1 The Problem of Tie Direction Inference

In this paper, we focus on the *tie direction inference* (TDI) problem for undirected social networks. Given an undirected social network denoted by a graph $G = \langle V, E\rangle$, where V is the set of individuals and E is the set of undirected social ties, the goal of TDI is to infer the corresponding hidden directed network, referred to as the *redirected network* denoted by $\vec{G} = \langle V, \vec{E}\rangle$, where $\vec{E}$ contains

[*] Corresponding authors: Chaokun Wang and Jianmin Wang.

WWW'14, April 7–11, 2014, Seoul, Korea.
ACM 978-1-4503-2744-2/14/04.
http://dx.doi.org/10.1145/2566486.2567968.



(a) Degree consistency (b) Triad status consistency (c) Similarity consistency (d) Collaborative consistency

Figure 1: The directionality patterns observed in directed social networks. (a) The *degree consistency* states that for a given directed tie $u \to v$, it's likely that u has lower in-degree but higher out-degree than v. (b) The *triad status consistency* assumes that given directed ties $u \to v$ and $v \to w$, u is more likely to propose the relationship with w than that in verse. (c) The *similarity consistency* says similar individuals have similar intentions in proposing relationships with others; in other words, given a directed tie $u \to v$, u is often more similar with the followers (a and b here) than followees (c and d here) of v, while v is more similar with the followees than followers of u (omitted in the figure for brevity). (d) The *collaborative consistency* implies that an individual u may have more common interests with her followees than her followers; for example, given a followee w and follower v of u, the followees (c and d) of u are usually more similar with the followees (e and f) of w than followees (a and b) of v.

the directed ties corresponding to the undirected ones in E. Specifically, given an undirected social tie (u, v), we need to distinguish the proposer and the responder.

The TDI problem is nontrivial and poses a set of unique challenges. First, are there any interesting patterns about the directionality in directed social networks? Second, for an undirected network which is totally unlabeled, can we infer the direction of each tie based on the network topology only? Third, when labeled data is available, can they be utilized easily to help our inference? Fourth, is there any simple way to construct pseudo but high-quality labeled data for supervision? Finally, whether the learned directed structure of an undirected network can be exploited to improve the performance of analysis or mining tasks upon the network?

1.2 Contributions

We engage in the investigation of the directionality patterns on the real-world directed social networks. Some interesting phenomena are observed (as exemplified in Fig. 1):

- The directed ties usually link from individuals with higher out-degrees but lower in-degrees to those with lower out-degrees but higher in-degrees.
- The directed ties tend to avoid loops.
- Similar individuals usually have similar positions (*proposer* or *responder*) in friendships with others.
- One usually has more common interests with her followees than her followers.

Based on the above observations, we develop an optimization framework, *ReDirect*, for inferring the directions of undirected social networks. It's designed with the following considerations: First of all, the learned directed network should be consistent with the directionality hypotheses, as observed in our real-world directed networks. Second, the learned directed network should improve the prediction performance in the predictive tasks than that on the original undirected network. Third, the framework should be general and can incorporate various predictive models according to the background. Last, the labeled data, when available, can be utilized to supervise the inference process. We achieve these by formulating the TDI problem using an optimization problem, and we're excited to find that using ReDirect the directions can be inferred with high accuracy based on the network topology only.

We conclude our main contributions as follows.

- We formulate the problem of tie direction inference (TDI) for undirected social networks. To the best of our knowledge, this is the first time the tie directionality is deeply studied.
- We investigate the directionality patterns in social networks and summarize our findings using four consistency hypotheses, including *degree consistency*, *triad status consistency*, *similarity consistency* and *collaborative consistency*.
- We present a general optimization framework, ReDirect, for solving the TDI problem based on the network topology only. We also demonstrate how ReDirect can be integrated with predictive models, using MF-based link prediction model as an example.
- We introduce supervision in the framework, and propose semi-supervised and self-supervised approaches for TDI. Furthermore, we propose a simple but effective way to build high-quality pseudo training data when labeled data is unavailable.
- We conduct comprehensive experiments on both directed and undirected social network datasets. Experimental results show that ReDirect can infer the directions of social ties effectively, and the learned directed network of the original undirected ones can benefit the predictive tasks.

1.3 Roadmap

The rest of the paper is organized as follows. We'll introduce the datasets and present our findings on these datasets firstly. Then we present the ReDirect framework for the TDI problem based on the proposed hypotheses in Sec. 3. In Sec. 4 we incorporate the supervision information into ReDirect and build semi-supervised and self-supervised approaches. The effectiveness of the proposed approaches is evaluated and demonstrated in Sec. 5. We illustrate the usefulness of ReDirect in link prediction in Sec. 6. In Sec. 7 we survey the related work and then conclude this study in Sec. 8.

2. DATA AND OBSERVATIONS

Before we go into the technical details, we first introduce our datasets, and the interesting patterns we observed on these datasets. Such patterns derive corresponding hypotheses which lay the foundation for our approach.

2.1 The Data

2.1.1 Directed Networks

We collected 6 directed networks for our experimental study.

- **Slashdot**. Slashdot allows users to tag each other as friends or foes. This dataset [18] consists of 77,360 users and 905,468 friend ties.
- **Epinions**. A who-trust-whom online social network of a general consumer review site Epinions.com. This dataset [30] consists of 75,879 users and 508,837 ties.
- **Tencent**. The Tencent Weibo network, one of the largest micro-blogging websites in China. This dataset is released

Table 1: The frequency that each event happens in directed social networks.

Dataset	Deg. Consis.						Triad Consis.		Simi. Consis.				Colla. Consis.	
	E1.1+	E1.1−	E1.2+	E1.2−	E1.3+	E1.3−	E2+	E2−	E3.1+	E3.1−	E3.2+	E3.2−	E4+	E4−
Slashdot	39.6%	60.2%	50.8%	48.4%	31.6%	68.0%	61.7%	38.3%	56.0%	44.0%	63.2%	36.8%	54.4%	43.3%
Epinions	63.2%	36.3%	72.5%	26.5%	47.5%	51.2%	83.4%	16.6%	58.3%	41.7%	78.6%	21.4%	52.7%	41.8%
Tencent	96.6%	3.0%	98.9%	0.8%	22.7%	71.6%	99.4%	0.6%	95.9%	4.1%	99.7%	0.3%	84.4%	9.6%
Sina	3.6%	96.4%	72.1%	17.3%	0.1%	99.9%	99.9%	0.1%	59.0%	41.0%	99.8%	0.2%	55.6%	31.8%
LiveJournal	48.9%	45.7%	59.0.5%	33.5%	35.7%	56.9%	78.4%	21.7%	57.6%	42.4%	98.6%	1.4%	57.8%	34.2%
Twitter	25.4%	74.6%	26.4%	73.1%	24.9%	75.1%	99.8%	0.2%	53.3%	46.7%	92.9%	7.1%	29.7%	62.8%

by KDD CUP 2012[1] and consists of 1,330,850 users and 5,064,496 ties.

- **Sina**. The Sina Weibo network, another of the largest microblogging websites in China. This dataset is from NLPIR[2] and consists of 528,390 users and 1,000,000 ties.
- **LiveJournal**. The LiveJournal online social network. This dataset [18] consists of 4,847,571 users and 68,993,773 ties.
- **Twitter**. A widely used microblogging system. The dataset [23] is comprised of 112,044 users, 468,238 following links among them.

2.1.2 Undirected Networks

We also collected 4 undirected networks.

- **CondMat**. The CondMat collaboration network in the domain of *condensed matter* collected from Arxiv[3]. This dataset [42] consists of 27,348 researchers and 72,119 co-author relationships.
- **HepEx**. The HepEx collaboration network in the domain of *high energy physics - experiment* collected from Arxiv. This dataset [42] consists of 5,667 researchers and 60,425 co-author relationships.
- **Flickr**. Flickr is a photo-sharing site based on a social network. The Flickr data [28] contains over 1,846,198 users and 22,613,981 ties.
- **Youtube**. YouTube is a popular video-sharing site that includes a social network. The YouTube data [28] consists of 1,157,827 users and 4,945,382 ties.

2.2 Hypotheses and Observations

In this section we study the patterns of directed ties in social networks, since a major motivation of our work is to find the underlying directionality patterns.

2.2.1 The Degree Consistency

The theory of status implies that in a directed tie the proposer usually views the responder as having higher status [17]. As the status cannot be measured explicitly, here we evaluate the correlation between the statuses and the degrees of the two participants of each directed tie. Specifically, for each directed tie $u \rightarrow v$, i.e. u is the proposer while v is the responder, as shown in Fig. 1(a), we test the frequency of following pairs of opposite events ("+" means the positive event while "−" means the negative):

- [E1.1+] The degree of u is lower than that of v;
- [E1.1−] The degree of u is higher than that of v;
- [E1.2+] The in-degree of u is lower than that of v;
- [E1.2−] The in-degree of u is higher than that of v;
- [E1.3+] The out-degree of u is lower than that of v;
- [E1.3−] The out-degree of u is higher than that of v.

In this study we don't consider the scenarios of equivalence as they are less informative.

The results are shown in Table 1. Firstly, we see neither E1.1+ nor E1.1− is prominent in our datasets. It suggests that the status is not necessarily consistent with the total degree, and the degree cannot be a reliable criterion for direction inference.

However, we find that E1.2+ appears far more than E1.2−, except on **Twitter**. Furthermore, the E1.3− is shown more prominent on all datasets than E1.3+. It leads us to naturally draw the conclusion that the statuses of individuals are positively correlated with in-degrees but negatively correlated with out-degrees.

We conclude the above observations with the *degree consistency hypothesis*: Given a directed tie $u \rightarrow v$ in a social network, u usually has higher out-degree but lower in-degree than v. In other words, individuals who receive many friendships tend to be responders, and those who propose many friendships tend to be proposers.

2.2.2 The Triad Status Consistency

Now we study the triad structures. Observing two directed relationships $u \rightarrow v$ and $v \rightarrow w$, as shown in Fig. 1(b), based on the status theory, we can learn that w usually has higher status than u. Thus the triad composed by u, v and w doesn't likely form a loop, otherwise the statuses of the participants cannot be partially ordered. To evaluate this, we test all the triads in directed network datasets and observe the following pair of events:

- [E2+] A triad in the directed network doesn't contain a loop;
- [E2−] A triad in the directed network contains a loop.

The frequencies of the above events are illustrated in Table 1, where we can see that the number of no-loop triads far exceeds that of the has-loop triads in all of the datasets. It can be naturally inferred that given directed relationships $u \rightarrow v$ and $v \rightarrow w$, u is more likely than w to propose the relationship between them. This is referred to as the *triad status consistency hypothesis*.

2.2.3 The Similarity Consistency

The theory of homophily [26] implies that the connected friends are usually similar. We extend this theory and assume that the similar individuals may have similar intentions in creating relationships with someone else. Equivalently, individuals with same position (proposer or responder) in relationships with others are likely similar. Taking the directed tie $u \rightarrow v$ shown in Fig. 1(c) as example, given that a, b and u are followers of v while c and d are followees, it can be assumed that u is more similar with a, b than c, d. We call this hypothesis *similarity consistency*.

We evaluate this hypothesis in the real-world datasets by analyzing the frequencies of following two pairs of opposite events, for each directed tie $u \rightarrow v$:

- [E3.1+] u is more similar with the followers of v;
- [E3.1−] u is more similar with the followees of v;
- [E3.2+] v is more similar with the followees of u;
- [E3.2−] v is more similar with the followers of u.

We extend the traditional Jaccard coefficient [21] and use the following measure to calculate the similarity of two individuals in directed social networks:

$$sim(u,v) = \frac{|F_{u\rightarrow} \bigcap F_{v\rightarrow}|}{|F_{u\rightarrow} \bigcup F_{v\rightarrow}|} \cdot \frac{|F_{u\leftarrow} \bigcap F_{v\leftarrow}|}{|F_{u\leftarrow} \bigcup F_{v\leftarrow}|}, \tag{1}$$

[1] http://www.kddcup2012.org/c/kddcup2012-track1

[2] http://www.nlpir.org

[3] Arxiv: http://arxiv.org/

where $F_{u\rightarrow}$ and $F_{u\leftarrow}$ denote the followees and followers of u, respectively.

From the results shown in Table 1 we can see that for both of E3.1 and E3.2 the positive events appear far more than the negative ones. This indicates that most of the directed ties satisfy the similarity hypothesis.

2.2.4 The Collaborative Consistency

The theory of homophily has been utilized in many predictive models as one's neighbors in social networks can provide additional information for modeling one's profiles. However, we believe that one's followers and followees play different roles in influencing one's behaviors, and therefore it's important and beneficial to distinguish them for such tasks. For an individual u, compared with her followers, her followees are expected to have more interests in common with u for that the relationships with them are proposed by u actively. We refer to this as the *collaborative consistency hypothesis*.

As the interests cannot be measured explicitly, here we consider one's preference for friends. As illustrated in Fig. 1(d), given an individual u, and her followee w and follower v, the followees (c and d) of u are assumed more similar with the followees (e and f) of w than those a and b of v. We justify this hypothesis by evaluating the following events for each individual u:

- [E4+] The followees of u are more similar with followees of her followees;
- [E4−] The followees of u are more similar with followees of her followers.

The results are illustrated in Table 1, which shows in most cases one share similar preference for friends with her followees than that with her followers. It verifies the above hypothesis and inspires us that by distinguishing one's followers and followees we can achieve better prediction performance for one's future behaviors.

3. THE FRAMEWORK

Given an undirected social network $G = \langle V, E\rangle$, the goal of the TDI problem is to infer the *redirected network* $\vec{G} = \langle V, \vec{E}\rangle$. Here we reformulate this problem in a probabilistic way. We define H as the adjacent matrix of G and $\vec{H}$ as the probabilistic adjacent matrix of $\vec{G}$. Each element in $\vec{H}$, say $\vec{H}_{i,j}$, corresponding to $H_{i,j}$ in G, represents the probability that i has proposed the relationship between i and j. Thus we can say $\vec{G}$ is an *$\vec{H}$-parameterized redirected network* of G. Now the goal of TDI is to find the optimal $\vec{H}$ which can explain the creation of each tie in G best.

The ReDirect framework is built based on the hypotheses discussed in Section 2.2, aiming at finding the redirected network which accords with these hypotheses as much as possible. Thus a key preliminary problem is how to measure the *consistency*, or equivalently, *inconsistency* of a network with these hypotheses.

In the remainder of this section, we'll first discuss how to measure the inconsistency of directed networks, and then introduce the ReDirect framework aiming at reducing the inconsistency most. An implementation based on matrix factorization is presented last.

3.1 Measuring the Inconsistency

In Section 2.2 we learned some directionality patterns from the directed social networks. Corresponding to the consistency hypotheses discussed previously, here we show how to measure the extent to which a network violates these hypotheses.

3.1.1 Measuring Degree Inconsistency

Firstly, we show how to measure the inconsistency of a directed network with the *degree consistency hypothesis*, which states that in directed networks the direction of a tie should be consistent with the in-degrees and out-degrees of the two participants. We define the *degree inconsistency* as follows:

Definition 1. The **degree inconsistency** of a directed network $\vec{G}$ parameterized by $\vec{H}$ is defined as:

$$\begin{aligned}\mathcal{C}_d(\vec{H}) = &\sum_{(i,j)\in\vec{E}} \Upsilon\left(\vec{H}_{i,j} - \vec{H}_{j,i}\ ,\ d_{in}(j) - d_{in}(i)\right)\\ &+ \sum_{(i,j)\in\vec{E}} \Upsilon\left(\vec{H}_{i,j} - \vec{H}_{j,i}\ ,\ d_{out}(i) - d_{out}(j)\right),\end{aligned} \tag{2}$$

where $\Upsilon(x, y)$ is a penalty function which requires x and y has same sign:

$$\Upsilon(x,y) = \begin{cases} 0 & \text{if sgn}(x) = \text{sgn}(y) \\ -\frac{x}{y} & \text{otherwise} \end{cases}, \tag{3}$$

and $d_{in}(i)$ and $d_{out}(i)$ refer to the in-degree and out-degree of i, respectively, defined as

$$\begin{aligned} d_{in}(i) &= \sum_{i'\in F_i} \vec{H}_{i',i}\ ,\\ d_{out}(i) &= \sum_{i'\in F_i} \vec{H}_{i,i'}\ ,\end{aligned} \tag{4}$$

and F_i is the set of friends of i, including both the followers and followees.

3.1.2 Measuring Triad Status Inconsistency

The hypothesis of *triad status consistency* assumes it's unlikely to form loops in triads. In other words, in a triad composed by i, m and n, the relative positions of i with m and n should be consistent with that between m and n. Specifically, we measure the status inconsistency in triads of a directed network quantitatively as follows:

Definition 2. The **triad status inconsistency** of a directed network $\vec{G}$ parameterized by $\vec{H}$ is defined as:

$$\mathcal{C}_t(\vec{H}) = \sum_{i\in V}\sum_{m,n\in F_i} \Upsilon\left(\vec{H}_{i,m} - \vec{H}_{i,n}, \vec{H}_{n,m} - \vec{H}_{m,n}\right). \tag{5}$$

The above definition is based on the following assumption. In a triad composed by i, m and n, if i has stronger intention in proposing a relationship with m than n (i.e. $\vec{H}_{i,m} > \vec{H}_{i,n}$), we can assume that m's status is higher than n's, and thus the intention of n in creating the relationship with m should be higher than that of m (i.e. $\vec{H}_{n,m} > \vec{H}_{m,n}$); vice versa.

3.1.3 Measuring Similarity Inconsistency

The third hypothesis, *similarity consistency*, supposes that two similar individuals should have similar intentions in making friends with someone else. The deviation of a network from this hypothesis can be measured by the *similarity inconsistency*, defined as follows:

Definition 3. The **similarity inconsistency** of a directed network $\vec{G}$ parameterized by $\vec{H}$ is defined as:

$$\mathcal{C}_s(\vec{H}) = \sum_{i\in V}\sum_{m,n\in F_i} \left(sim(m,n)\cdot(\vec{H}_{i,m} - \vec{H}_{i,n})^2\right)^{s(m,n)} \tag{6}$$

where $sim(m, n)$ measures the similarity between m and n, $s(m, n)$ is an indicator function defined as

$$s(m,n) = \begin{cases} 1 & \text{if } sim(m,n) \geq \alpha \\ -1 & \text{otherwise} \end{cases}, \tag{7}$$

and α is the similarity threshold. In our settings, α for each individual i takes the average similarity among the friends of i.

We explain this definition intuitively as follows. Two individuals, m and n are regarded *similar* if their similarity is greater than the given threshold; otherwise they are *dissimilar*. In the above definition, for *similar* neighbors m and n of i, a smaller difference between $\vec{H}_{i,m}$ and $\vec{H}_{i,n}$ makes it more consistent with the similarity consistency hypothesis. On the contrary this hypothesis requires larger difference for the *dissimilar* neighbors.

3.1.4 Measuring Collaborative Inconsistency

The *collaborative consistency hypothesis* assumes that one's followees reveal more about her interest than her followers. That means, by identifying the directions towards one's neighbors and distinguishing one's followees and followers, we can infer one's profiles and interests better. Thus here we measure this inconsistency of a given directed network using the performance of link prediction models upon the network. We define $f_{\vec{H}}(i,j)$ as the prediction function, which gives a score indicating the possibility that i will propose a link to j in the future, based on the predictive model built upon $\vec{H}$.

Definition 4. The **collaborative inconsistency** of a directed network $\vec{G}$ parameterized by $\vec{H}$ is defined as:

$$\mathcal{C}_c(\vec{H}) = \sum_{i\in V} \sum_{d\in D_i^+, l\in D_i^-} (f_{\vec{H}}(i,l) - f_{\vec{H}}(i,d))^2, \tag{8}$$

where D_i^+ is the new friends set of i in the future, and D_i^- is the set of non-friends.

This definition is based on the assumption that if $\vec{H}$ could depict the nature of the network well, the link prediction model built upon the network should be more accurate. The implementation of the collaborative inconsistency measure can vary with the implementation of the prediction model f.

3.2 The ReDirect Framework

Given the above measures, we can solve our problem using an optimization framework, aiming at finding an optimal $\vec{H}$ which violates the hypotheses least. We define the loss function as

$$\begin{aligned} L(\vec{H}) = \lambda_d \mathcal{C}_d(\vec{H}) + \lambda_t \mathcal{C}_t(\vec{H}) \\ + \lambda_s \mathcal{C}_s(\vec{H}) + \lambda_c \mathcal{C}_c(\vec{H}), \end{aligned} \tag{9}$$

thus we build the $ReDirect$ optimization framework as:

$$\begin{aligned} \vec{H}^* = \arg\min_{\vec{H}} \quad & L(\vec{H}) \\ s.t. \quad & \vec{H}_{i,j} + \vec{H}_{j,i} = 1, \forall (i,j) \in E \\ & 0 \le \vec{H}_{i,j} \le 1, \forall (i,j) \in E\ . \end{aligned} \tag{10}$$

We can transform the constrained optimization problem into an unconstrained one using a regularizer :

$$\Omega(\vec{H}) = \sum_{(i,j)\in E} \left((\vec{H}_{i,j} + \vec{H}_{j,i} - 1)^2 + \Gamma(\vec{H}_{i,j}) + \Gamma(1 - \vec{H}_{i,j}) \right), \tag{11}$$

where the penalty function Γ is defined as

$$\Gamma(x) = \begin{cases} x^2 & \text{if } x < 0 \\ 0 & \text{otherwise} \end{cases}, \tag{12}$$

thus the objective function of the final unconstrained model is:

$$\begin{aligned} L(\vec{H}) = \lambda_d \mathcal{C}_d(\vec{H}) + \lambda_t \mathcal{C}_t(\vec{H}) \\ + \lambda_s \mathcal{C}_s(\vec{H}) + \lambda_c \mathcal{C}_c(\vec{H}) + \mu\Omega(\vec{H}). \end{aligned} \tag{13}$$

Depending on how the link prediction model f is designed, we can build various implementations of *ReDirect*. Next, we introduce one based on matrix factorization as an example.

3.3 An MF-based Implementation

Matrix factorization (MF) [12], as a popular technique of collaborative filtering, has been successful in many prediction or recommendation tasks such as item recommendation [32] and link prediction [27]. It learns the latent features for each user by factorizing the adjacent matrix H using $P, Q \in \mathbb{R}^{k\times n}$:

$$H \approx P^T Q, \tag{14}$$

where n and k are the numbers of users and latent features, respectively. Each column $\mathbf{p}_i$ and $\mathbf{q}_i$ in the latent feature matrices P and Q is a latent feature vector corresponding to user i, indicating the intention of user i for making new friends and the preference of user i for accepting new friends, respectively. The link prediction can be formulated as a matrix completion problem using the learned feature matrices.

Here we assume the objective matrix $\vec{H}$ can be decomposed using P and Q. Based on the learned feature matrices we define the prediction function f as:

$$f_{\vec{H}}(i,j) = \vec{H}_{i,j} = \mathbf{p}_i^T \mathbf{q}_j. \tag{15}$$

By introducing the above equation, the inconsistency measures defined in Sec. 3.1 can be rewritten using P and Q as follows:

$$\begin{aligned} \mathcal{C}_d(P,Q) = & \sum_{(i,j)\in\vec{E}} \Upsilon\left(\mathbf{p}_i^T\mathbf{q}_j - \mathbf{p}_j^T\mathbf{q}_i\ ,\ d_{in}(j) - d_{in}(i)\right) \\ & + \sum_{(i,j)\in\vec{E}} \Upsilon\left(\mathbf{p}_i^T\mathbf{q}_j - \mathbf{p}_j^T\mathbf{q}_i\ ,\ d_{out}(i) - d_{out}(j)\right), \end{aligned} \tag{16}$$

$$\mathcal{C}_t(P,Q) = \sum_{i\in V} \sum_{m,n\in F_i} \Upsilon\left(\mathbf{p}_i^T\mathbf{q}_m - \mathbf{p}_i^T\mathbf{q}_n\ ,\ \mathbf{p}_n^T\mathbf{q}_m - \mathbf{p}_m^T\mathbf{q}_n\right), \tag{17}$$

$$\mathcal{C}_s(P,Q) = \sum_{i\in V} \sum_{m,n\in F_i} \left(sim(m,n) \cdot \left(\mathbf{p}_i^T\mathbf{q}_m - \mathbf{p}_i^T\mathbf{q}_n\right)^2 \right)^{s(m,n)}, \tag{18}$$

$$\mathcal{C}_c(\vec{H}) = \sum_{i\in V} \sum_{d\in D_i^+, l\in D_i^-} \left(\mathbf{p}_i^T\mathbf{q}_l - \mathbf{p}_i^T\mathbf{q}_d\right)^2. \tag{19}$$

Thus, the model defined in Eq. 10 can be reformulated as:

$$\begin{aligned} P^*, Q^* = \arg\min_{P,Q} \quad & L(P,Q) \\ s.t. \quad & \mathbf{p}_i^T\mathbf{q}_j + \mathbf{p}_j^T\mathbf{q}_i = 1, \forall (i,j) \in E \\ & 0 \le \mathbf{p}_i^T\mathbf{q}_j \le 1, \forall (i,j) \in E \end{aligned} \tag{20}$$

whose objective function is:

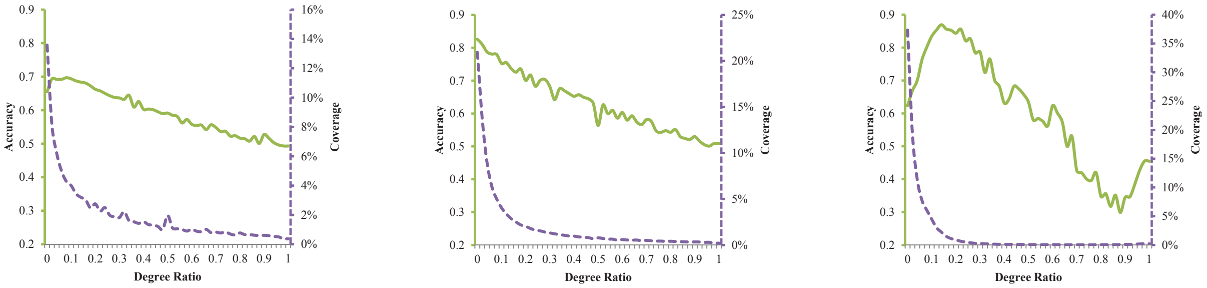
$$\begin{aligned} L(P,Q) = \lambda_d \mathcal{C}_d(P,Q) + \lambda_t \mathcal{C}_t(P,Q) \\ + \lambda_s \mathcal{C}_s(P,Q) + \lambda_c \mathcal{C}_c(P,Q) \end{aligned} \tag{21}$$

By introducing the following regularizer:

$$\Omega(P,Q) = \sum_{(i,j)\in E} \left((\mathbf{p}_i^T\mathbf{q}_j + \mathbf{p}_j^T\mathbf{q}_i - 1)^2 + \Gamma(\mathbf{p}_i^T\mathbf{q}_j) + \Gamma(1 - \mathbf{p}_i^T\mathbf{q}_j) \right), \tag{22}$$

we transform the constrained optimization problem into an unconstrained one whose the objective function is:

$$\begin{aligned} L(P,Q) = \lambda_d \mathcal{C}_d(P,Q) + \lambda_t \mathcal{C}_t(P,Q) \\ + \lambda_c \mathcal{C}_c(P,Q) + \lambda_s \mathcal{C}_s(P,Q) + \mu\Omega(P,Q). \end{aligned} \tag{23}$$



(a) Epinions (b) Tencent (c) Twitter

Figure 2: The correlation between the direction and the degree ratio.

This model can be solved by gradient descent. We omit the details due to the space limit. With the learned P and Q, we can get the optimal probabilistic adjacent matrix $\vec{H} = P^T Q$, which can be used to estimate the hidden directions for the original network G. For each pair of connected nodes i and j in G, if $\vec{H}_{i,j} - \vec{H}_{j,i} > \sigma$, where σ is a confidence threshold between $(0, 1)$, we say i proposed the relationship with j; on the contrary, j is assumed as the proposer of the relationship if $\vec{H}_{j,i} - \vec{H}_{i,j} > \sigma$. When $|\vec{H}_{i,j} - \vec{H}_{j,i}| \leq \sigma$, we say i and j created the relationship together.

4. INTRODUCING SUPERVISION

ReDirect is an unsupervised optimization framework for learning the hidden directions of undirected social networks based on the structural patterns we observed in real-world networks. By restraining the inferred network using consistency hypotheses, we try to learn the hidden/original directed network which can explain the nature of the visible undirected network without any training data. However, in situations where the directions of some links are available, we believe such information can guide our inference for the remaining links. In this section, we show how to incorporate the partially labeled data in ReDirect to build a semi-supervised model. We'll also discuss when no labeled data is in existence, how we can also construct the pseudo-labeled data with simple but effective heuristics and use the model in a self-supervised manner.

4.1 Semi-Supervised ReDirect

For the given undirected social network $G = \langle V, E \rangle$, where E is the set of all links, we assume $\vec{E}'$ is the set of labeled directed ties of G. For each directed tie $(i \to j) \in \vec{E}'$, we should assure that $\vec{H}_{i,j} - \vec{H}_{j,i} > \sigma$ after the model learning. We define the following loss function for the labeled data:

$$L_v(\vec{H}) = \sum_{(i \to j) \in \vec{E}'} \left(\vec{H}_{i,j} - \vec{H}_{j,i} - \sigma \right)^{2 \cdot h(i,j)} , \tag{24}$$

where $h(i, j)$ is an indicator function defined as

$$h(i,j) = \begin{cases} 1 & \text{if } \vec{H}_{i,j} - \vec{H}_{j,i} - \sigma < 0 \\ 0 & \text{otherwise} \end{cases} . \tag{25}$$

By adding this with the original loss function, we incorporate the known information in the original unsupervised approach, and get the semi-supervised ReDirect model, whose objective function is defined as:

$$L_{semi}(\vec{H}) = L_v(\vec{H}) + L(\vec{H}) . \tag{26}$$

This model can be effective because the labeled ties will propagate the knowledge to the remainder of the network via consistency constraints. Thus even only few data is available, they can also benefit the model learning via knowledge propagation.

4.2 Self-Supervised ReDirect

While the semi-supervised ReDirect allows us to supervise the direction inference using the labeled information, we also notice that such information is unavailable in most undirected social networks. Thus we consider an alternative way: Can we construct some pseudo-labeled data based on the observed patterns?

In Sec. 2.2.1, we analyzed the macroscopic correlation between the degree and the direction of a link. The direction is not strongly correlated with the degrees of the two endpoints of a link, but correlated with the in-degrees and the out-degrees. Unfortunately, we cannot distinguish the in-degree and out-degree in undirected networks. Though the degree cannot be a reliable feature for judging the directions for *all ties*, we assume it may be useful for *some ties*. For an undirected tie (i, j), if the degree of j far exceeds that of i, it may be more possible that i proposed the relationship with j. Thus we're interested in that whether there exists correlation between the direction and the ratio of the degrees of the two participants of a tie.

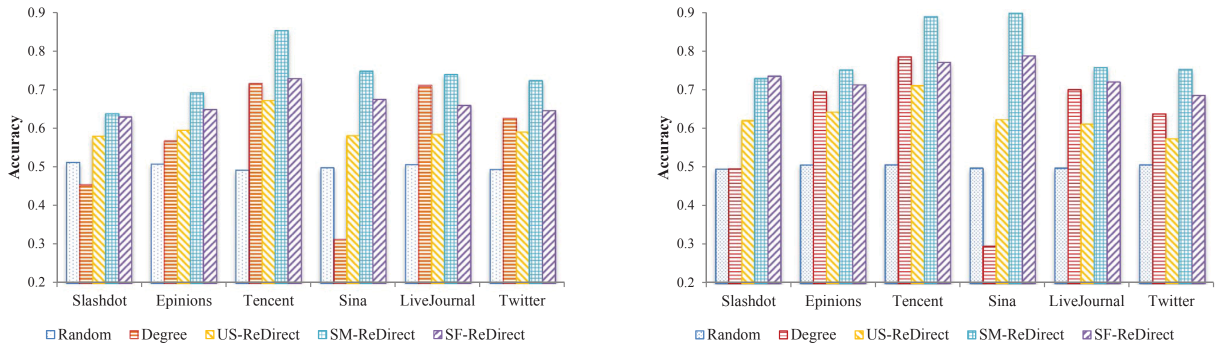
We evaluate this on our real-world datasets. For each pair of friends i and j in a given social network, we assume i has less degree than j. We calculate the *degree ratio* $deg(i)/deg(j)$ for each pair in the network. If the real direction in the given network is $i \to j$, we say it's a positive instance; otherwise it's negative. According to this calculation method, the ratio should take values between 0 and 1. We split this region into multiple intervals. For each interval, we measure the *accuracy*, which means the percentage of positive instances among the ties that fall in this interval, and the *coverage*, which means the percentage of positive instances that fall in this interval among all positive instances.

Fig. 2 illustrates the results on three of our datasets. Both the accuracy and coverage are high when degree ratio is low. Similar results are observed on other datasets. This convinces us that there's an increasing probability that one proposes a relationship with another who has much higher degree than the proposer.

This finding inspires us to build pseudo-labeled data in a simple but effective way using these pairs of nodes with high degree difference. For each pair of connected nodes i and j, we calculate their degree ratio. If the ratio is less than some threshold, we add the directed edge $i \to j$ to the labeled edge set $\vec{E}'$. In practice when it's difficult to set a threshold, we can sort these pairs by degree ratio in descending order and select the top-*K* (percent) of pairs as the training data.

5. EXPERIMENTAL EVALUATION

In this section, we evaluate the effectiveness of proposed methods for tie direction inference. We'll first describe how we prepare the dataset, and then present the experimental results to compare



(a) Performance comparison for all edges

(b) Performance comparison for edges involving egos

Figure 3: The overall performance of social tie direction inference on each dataset.

the performance of our methods with baselines. At last the effects of the consistency constraints and supervision are discussed.

5.1 Data Preparation

We conducted experiments on six directed social networks introduced in Sec. 2.1.1. Since the original networks are too large and sparse, we sampled subnetworks, each of which contains nearly 50,000 nodes, from original networks using breadth-first traversal. During this process, the nodes whose whole neighbors had been added in the sampled network are called *egos*. Furthermore, for each individual i, 10% of her neighbors are extracted as her *new friends* D_i^+, and the 2-hop neighbors are regarded as *non-friends* D_i^-.

For each directed network in our datasets we transform it to undirected by removing the directions of all ties. We apply tie direction inference methods on the obtained undirected networks and evaluate their performance with all the unidirectional ties in the original directed networks as ground-truth.

5.2 Performance Comparison

We evaluated the accuracy of direction inference of the proposed approaches and baselines on the above datasets. In our experiments, we tested three variants of our proposed approach:

- The original *Unsupervised ReDirect* (US-ReDirect) without supervision;
- The *Semi-supervised ReDirect* (SM-ReDirect). For each dataset we extracted 20% of directed ties as training data and the remaining for testing;
- The *Self-supervised ReDirect*(SF-ReDirect). For each dataset, we calculated the degree ratio of each tie, and the top 20% instances with least degree ratio were considered as labeled training data.

We compared our methods with two baselines:

- *Random*, which decides the direction randomly;
- *Degree*, which decides the direction based on the degree, and regards the one with lower degree as the proposer.

We set the *confidence threshold* using extreme value $\sigma = 0$ to evaluate the performance of the above methods. The number of latent features k is set 10 in ReDirect.

The results are shown in Fig. 3. Fig. 3(a) shows the accuracy for all the ties in the network, while Fig. 3(b) shows the accuracy for the ties involving egos, whose neighbors have been all contained.

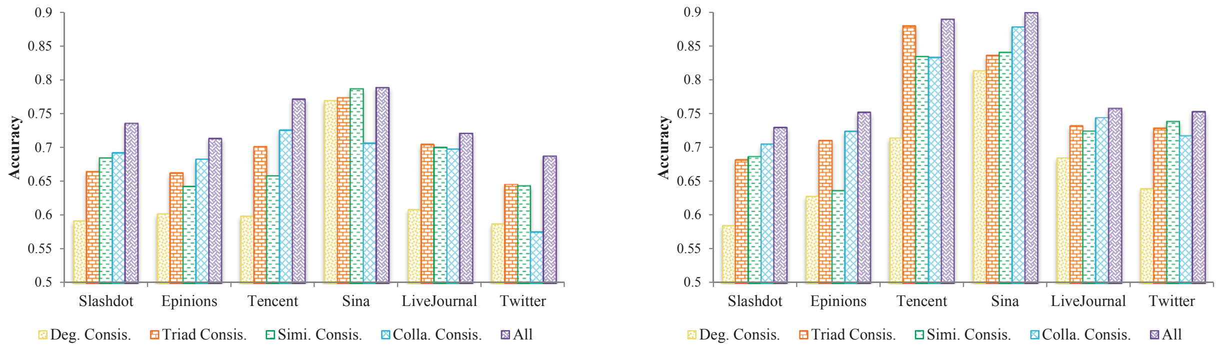
We can see our proposed methods outperform the baselines on most datasets. The *Random* method achieves nearly 50% accuracy in the experiments. The *Degree*-based method shows high variance on the datasets. From Fig. 3(a), we can see it outperforms the *US-ReDirect* on **Tencent**, **LiveJournal** and **Twitter**, but shows even worse performance than *Random* on **Slashdot** and **Sina**. This result coincides with the finding in Sec. 2.2.1 and 3.3, revealing that the degree may help us to analyze the possible directions of ties, but it's not reliable. On the contrary, our proposed methods show stable performance on all the datasets. *SM-ReDirect* performs best on all of the datasets. Even only 20% data is used for training, it increases the performance of *US-ReDirect* significantly. This proves that our semi-supervised learning approach can propagate important knowledge about the network structure and guide the direction learning within the framework constrained by the consistency hypotheses. Though worse than the semi-supervised approach, *SF-ReDirect* always outperforms the original unsupervised method, showing the effectiveness of the pseudo training data selected according to the degree ratio.

Furthermore, comparing Fig. 3(b) with Fig. 3(a), we can find our methods achieve better performance for the ties involving egos than others. For the egos we extracted all her neighbors and get a full picture of her network structure. This enables us to get more sufficient knowledge about the interests and patterns of egos, and thus make more accurate inference.

5.3 Effectiveness of Consistency Constraints

In Sec. 2.2, we proposed 4 consistency hypotheses based on our observations on real-world data. These hypotheses form the basis of the proposed ReDirect framework. In this subsection, we evaluate the effectiveness of each consistency constraint in experiments by removing other constrains from the original model. Here we only show our experiments for *SF-ReDirect* and *SM-ReDirect* on three datasets for the limit of space.

In Fig. 4(a), we can see the effectiveness of each constraint in *SF-ReDirect*. We find the *degree consistency constraint* shows worst performance on most datasets. This conveys that, though the in-degree and out-degree patterns are shown significant in real-world directed networks, we cannot rebuild the directed networks only by optimizing the in-degree and out-degree allocation for the given undirected network without other information. The other three constraints show varying effectiveness on the datasets. *Similarity consistency* shows best performance than others on **Sina**, and *triad status consistency* outperforms others on **LiveJournal** and **Twitter**, while *collaborative consistency* wins on **Slashdot**, **Epinions** and **Tencent**. Similar results are observed on the experiments for *SM-ReDirect*, as shown in Fig. 4(b), where *collaborative consistency* outperforms others on 4 of 6 datasets. Generally speaking, the *collaborative consistency* is considered as more important because it learns the structure based on its predictive power for the



(a) Self-supervised ReDirect

(b) Semi-supervised ReDirect

Figure 4: The effect of each consistency constraint in self-supervised ReDirect and semi-supervised ReDirect on each dataset. Here we show the accuracy of direction inference for the ties involving egos.

future, while others are necessary for constraining the learned network structure. We see the whole ReDirect with all the constraints included shows best performance.

5.4 Effectiveness of Supervision

In previous experiments, we show the training data can improve the performance of ReDirect significantly. In this subsection we investigate the effect of supervision for ReDirect by varying the amount of training data.

In *SF-ReDirect*, we selected the top-K (percent) edges with minimal degree ratios as positive instances for training. Fig. 5 shows how the performance of *SF-ReDirect* changes with different amount of training data. At the first beginning, the performance increases while adding more training data, but decreases when more data is regarded as labeled. This is consistent with the observation in Sec. 3.3 because when we increase K, we make more errors in the pseudo-positive instances selection, and thus introduce more noise to our model.

Another interesting phenomenon is that the accuracy for ties involving egos decreases faster than that for all ties. One possible reason is that the egos have more complete neighbors and thus suffer more from the increasing amount of noisy training data. The error information will propagate around the egos via the consistency constraints, and leads to a *consistent* but *wrong* structure.

Unlike *SF-ReDirect*, *SM-ReDirect* benefits more from supervision when more training data becomes available, as shown in Fig. 6.

6. APPLICATION IN PREDICTIVE TASKS

The social network has been proved beneficial in many predictive tasks [21, 31, 20, 24].In this section, we show how ReDirect can improve the performance of traditional predictive models upon social networks by investigating the case of link prediction.

6.1 Link Prediction

The link prediction problem [21] can be formulated as follows. Given a snapshot of a social network $G = \langle V, E\rangle$ at a particular time and a pair of candidate users (non-friends) i and j, how is it possible that i and j become friends in the future?

Most work in undirected social networks assumes the network is plain and ignores its directionality nature. In this section, we study whether the inferred directions can improve the link prediction performance. It should be noted that we don't intend to propose new link prediction methods. We just evaluate to what extent the redirected network inferred by ReDirect can improve the performance of tradition predictive methods.

Here we consider four popular methods:

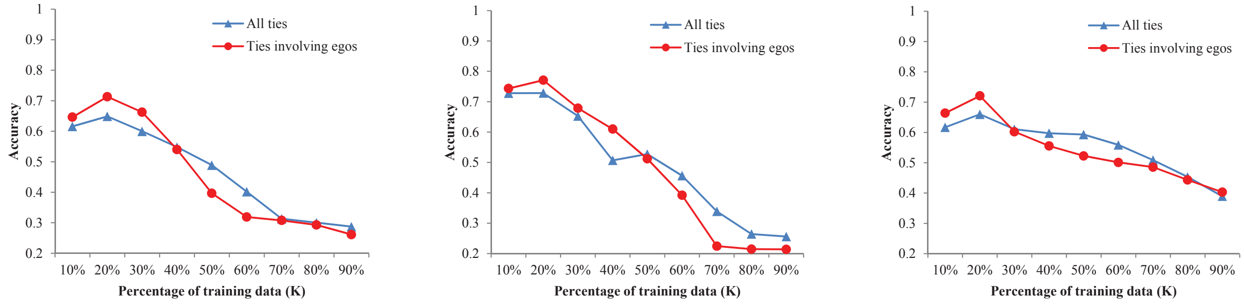
- *Common Friends*, which measures the proximity of nodes using the number of their common friends [21]. Here we reformulate this approach using the adjacent matrix H of the network G. For directed network, the prediction function for a directed link $i \to j$ can be defined as:
$$f_{cmf}(i \to j) = H_{i,:} \cdot H_{:,j} \ . \tag{27}$$
- *Jaccard Coefficient*, another popular method based on the Jaccard Coefficient [21]. Using adjacent matrix H, we define the prediction function as:
$$f_{jac}(i \to j) = \frac{sum(H_{i,:} \cdot H_{:,j})}{sum(H_{i,:}) + sum(H_{:,j})} \ . \tag{28}$$
- *Exponential Kernel*, which measures the node proximity using exponential kernels, defined as [11] ($\beta = 0.8$ here) :
$$\begin{aligned} f_{ker}(i \to j) = e^{\beta H} &= \lim_{n\to\infty} (1 + \frac{\beta H}{n})^n \\ &= I + \beta H + \frac{\beta^2}{2!}H^2 + \frac{\beta^3}{3!}H^3 + \cdots \ . \end{aligned} \tag{29}$$
- *Matrix Factorization*. As stated in Sec. 3.3, we factorize the adjacent matrix $H \approx P^T Q$ and measure the proximity in the latent feature space defined by P and Q, as
$$f_{mf}(i \to j) = \mathbf{p}_i^T \mathbf{q}_j \ . \tag{30}$$

Furthermore, for an undirected candidate link (i, j) in undirected networks, the prediction function is usually defined as:
$$f(i, j) = f(i \to j) + f(j \to i) \ . \tag{31}$$

6.2 Experiments on Directed Networks

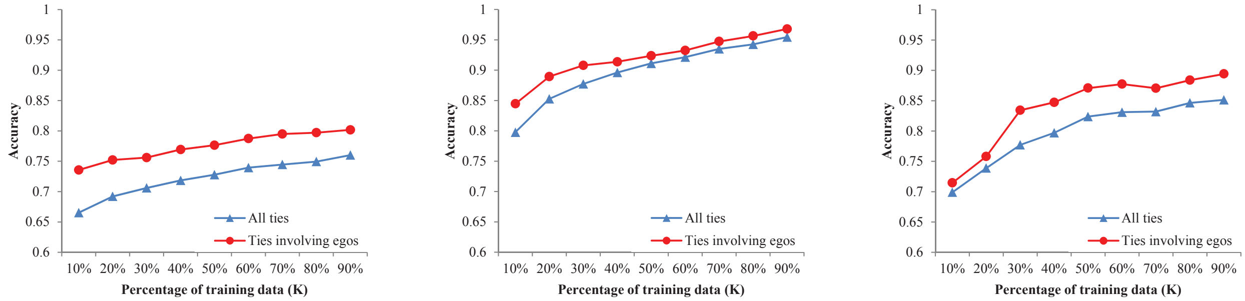
We first evaluated the effect of ReDirect for the link prediction on real-world directed network datasets. For each dataset, we extracted all individuals and 80% of links among them to build an experimental directed network $G^{(d)}$. We constructed the testing set for link prediction by considering the 2-hop neighbors of each individual u in $G^{(d)}$: Those who appear as friends of u in original networks are regarded as positive instances while others as negative. For each dataset, we consider the following *derived* networks:

- The original *directed network* $G^{(d)}$.
- The *undirected network* $G^{(u)}$ built by removing the directions of the links from the directed network $G^{(d)}$.
- The *us-redirected network* $G^{(rd)}_{us}$, i.e. the inferred directed network from $G^{(u)}$ using unsupervised ReDirect.



(a) Epinions (b) Tencent (c) LiveJournal

Figure 5: The variance of social tie direction inference performance with the amount of training data in self-supervised ReDirect.



(a) Epinions (b) Tencent (c) LiveJournal

Figure 6: The variance of social tie direction inference performance with the amount of training data in semi-supervised ReDirect.

- The *sf-redirected network* $G_{sf}^{(rd)}$, i.e. the inferred directed network from $G^{(u)}$ using self-supervised ReDirect.
- The *sm-redirected network* $G_{sm}^{(rd)}$, i.e. the inferred directed network from $G^{(u)}$ using semi-supervised ReDirect.

Then we applied the above link prediction methods on all the above *derived* networks, instead of using the original *directed network* $G^{(d)}$ only. We evaluated the performance of link prediction using AUC (Area Under the ROC Curve).

The experimental results are shown in Fig. 7. Firstly, we see that for most datasets and most link prediction methods, prediction based on the original *directed networks* $G^{(d)}$ achieves better performance than that based on the *undirected networks* $G^{(u)}$. This illustrates that much information is lost during converting the original directed network to undirected. Though many networks appear as undirected, we believe the directions are still existing but just hidden. That's why we're motivated to recover the hidden directions from undirected networks.

Furthermore, we can find that the *redirected networks* can improve the link prediction than the *undirected networks* $G^{(u)}$, which demonstrates the ability of *ReDirect* for recovering the original directed networks. Moreover, the *redirected networks* even show better performance than the original *directed networks* $G^{(d)}$ in some algorithms and some datasets, such as on all datasets using *Matrix Factorization*, on **Epinions** and **LiveJournal** using *Jaccard Coefficient* and *Exponential Kernel*. This shows that the *redirected networks* not only recover the hidden directions, but also encode more precise collaborative effects in the probabilistic matrix $\vec{H}$.

On directed networks, the performance of *sm-redirected networks* $G_{sm}^{(rd)}$ is usually better than *sf-redirected networks* $G_{sf}^{(rd)}$, while the latter shows better performance than *us-redirected networks* $G_{us}^{(rd)}$. This is consistent with the previous experiments and conclusions that the semi-supervised and self-supervised ReDirect can recover the directions better than the unsupervised approach while the semi-supervised one performs best.

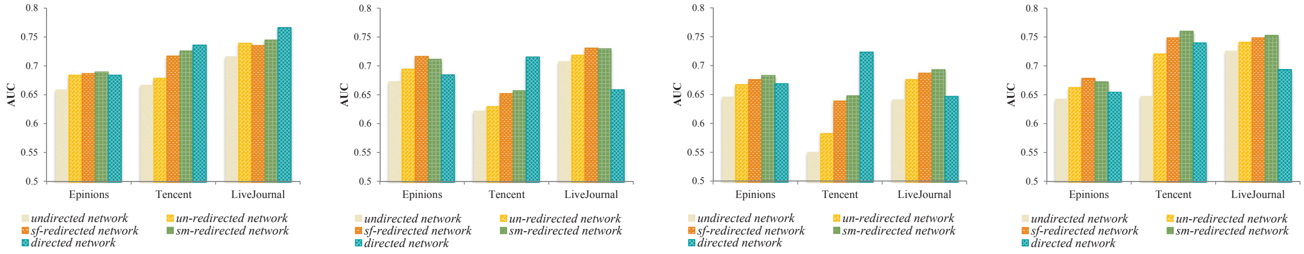
6.3 Experiments on Undirected Networks

Next we tested the algorithms on four real-world undirected network datasets. For each dataset, we picked 80% of existing links to build an experimental undirected network $G^{(u)}$ and constructed the positive and negative instances for testing in the similar way with the experiments on directed network datasets. As for $G^{(u)}$ we don't have the labeled directions, we only evaluated the algorithms on the following three *derived* networks for each dataset:

- The original *undirected network* $G^{(u)}$.
- The *us-redirected network* $G_{us}^{(rd)}$, i.e. the inferred directed network from $G^{(u)}$ using unsupervised ReDirect.
- The *sf-redirected network* $G_{sf}^{(rd)}$, i.e. the inferred directed network from $G^{(u)}$ using self-supervised ReDirect.

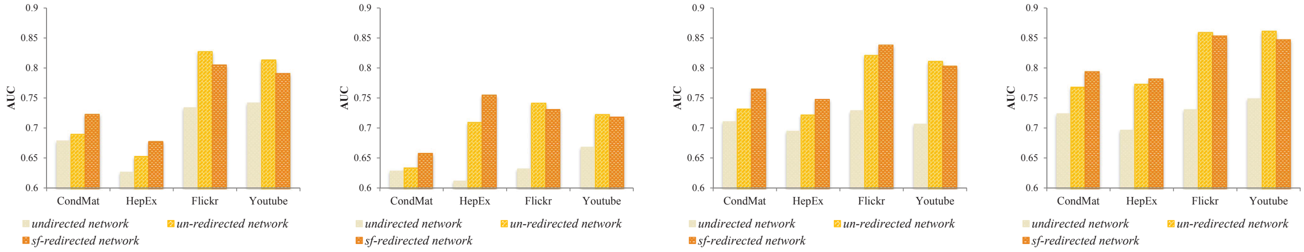
Similar with the experiments on directed networks, we see the *redirected networks* can improve the link prediction performance significantly on all datasets and for all algorithms. This convinces us that ReDirect doesn't rely on any specific algorithms; on the contrary, it can be used to preprocess the network data and benefit other network analysis or mining tasks.

Unlike the experimental results on directed network datasets, the *sf-redirected networks* are not always better than the *us-redirected networks* in experiments with undirected ones. It's reasonable because the performance of self-supervised ReDirect relies on the quality of the self-selected training data based on the degree ratios. As shown in Sec. 5.4, various thresholds for selecting the training



(a) *Common Friends* (b) *Jaccard Coefficient* (c) *Exponential Kernel* (d) *Matrix Factorization*

Figure 7: The performance of variant link prediction methods on each directed social network dataset.



(a) *Common Friends* (b) *Jaccard Coefficient* (c) *Exponential Kernel* (d) *Matrix Factorization*

Figure 8: The performance of variant link prediction methods on each undirected social network dataset.

data can lead to quite different results. This points out one future direction that we can study how to learn the labels by itself more intelligently and accurately.

7. RELATED WORK

In this section we survey lines of related study.

The nature of social ties. The social ties are born with plenty of properties. One of them is the strength of ties [7], which has attracted many attentions in recent years. While earlier work focused on distinguishing the strong and weak ties by formulating this as a binary classification problem [6, 10], most work studied how to quantificationally measure the strength of social ties recently [35, 1, 43]. Another important property of a social tie is its sign, i.e. "positive" or "negative". Researchers have studied the structures and properties of signed networks [17, 15, 14]. Furthermore, many researchers focused on the sign classification of edges [13, 16, 36, 38], while others studied the signed link prediction problem on signed networks [3, 4]. Except the strength and sign, other properties have also been studied, such as the type of tie (family, colleague, classmates, etc) [33] and the role of the participants (competitive relationship [37], advisor-advisee relationship [34]).

Unlike existing work, we focus on the directionality of undirected social ties in this paper. To the best of our knowledge, it is the first time that this problem is proposed and studied.

Link prediction. This is another sort of work related with this paper, which has become a hot topic since the seminal work of Liben-Nowell and Kleinberg [21]. The basic method is based on the local neighborhood structures, as surveyed by Liben-Nowell and Kleinberg [21]. Another kind of popular approach utilizes the random walk to measure the node proximity on the whole network [19, 2]. Furthermore, matrix factorization methods [27, 5], content-based methods [8, 25], attribute-based methods [39] and behavior modeling based methods [41, 42, 40] have been studied respectively. Supervised methods are also investigated [9, 22].

However, most of existing approaches for link prediction ignore the nature of directionality. In this paper we try to recover the hidden directions of undirected networks and show that the learned directed network can benefit the link prediction tasks, regardless of which model is used.

8. CONCLUSION AND FUTURE WORK

In this paper, we argue that the directionality is a hidden but important property of undirected social ties and propose the TDI (tie direction inference) problem for the first time. Based on the observations on the real-world directed social networks, we propose four consistency hypotheses to describe the observed directionality patterns, and build the *ReDirect* framework to recover the hidden directions based on that. We highlight ReDirect in following aspects: First of all, it achieves fantastic performance based on the network topology only, without any external information. Moreover, it's a general framework which can incorporate any predictive models, though we provide an implementation using MF for link prediction in this paper as example. Furthermore, utilizing the labeled or self-labeled data, the ReDirect framework can incorporate the semi-supervision or self-supervision easily and improve the performance significantly. Besides, ReDirect can benefit other analysis or mining tasks on social networks with the learned directed network. The effectiveness of ReDirect for direction inference and link prediction has been proved in our experimental study.

This is just the first step towards directionality study and we can see many directions for future work. Among them is to incorporate other predictive models in ReDirect and evaluate their performance. The usefulness of the learned redirected networks is also expected to be evaluated in other tasks, including item recommendation, spam detection, etc. Furthermore, we can also study other heuristics for labeling the high-confidence pseudo self-training data.

9. ACKNOWLEDGMENTS

This work is supported in part by the National Natural Science Foundation of China (No. 61373023, No. 61170064, No. 61133002), the National High Technology Research and Development Program of China (No. 2012AA011002).

10. REFERENCES

[1] C. Au Yeung and T. Iwata. Strength of social influence in trust networks in product review sites. In *WSDM '11*, 2011.

[2] L. Backstrom and J. Leskovec. Supervised random walks: predicting and recommending links in social networks. In *WSDM '11*, 2011.

[3] K.-Y. Chiang, N. Natarajan, A. Tewari, and I. S. Dhillon. Exploiting longer cycles for link prediction in signed networks. In *CIKM '11*, 2011.

[4] T. DuBois, J. Golbeck, and A. Srinivasan. Predicting trust and distrust in social networks. In *SocialCom/PASSAT'11*, 2011.

[5] D. M. Dunlavy, T. G. Kolda, and E. Acar. Temporal link prediction using matrix and tensor factorizations. *ACM Trans. Knowl. Discov. Data*, 5(2):10:1–10:27, 2011.

[6] E. Gilbert and K. Karahalios. Predicting tie strength with social media. In *CHI '09*, 2009.

[7] M. Granovetter. The Strength of Weak Ties. *The American Journal of Sociology*, 78(6):1360–1380, 1973.

[8] J. Hannon, M. Bennett, and B. Smyth. Recommending twitter users to follow using content and collaborative filtering approaches. In *RecSys '10*, 2010.

[9] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *SDM '06 workshop on Link Analysis, Counterterrorism and Security*, 2006.

[10] I. Kahanda and J. Neville. Using transactional information to predict link strength in online social networks. In *ICWSM'09*, 2009.

[11] R. I. Kondor and J. D. Lafferty. Diffusion kernels on graphs and other discrete input spaces. In *ICML '02*, 2002.

[12] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[13] J. Kunegis, A. Lommatzsch, and C. Bauckhage. The slashdot zoo: mining a social network with negative edges. In *WWW '09*, 2009.

[14] J. Kunegis, J. Preusse, and F. Schwagereit. What is the added value of negative links in online social networks? In *WWW '13*, 2013.

[15] J. Kunegis, S. Schmidt, A. Lommatzsch, J. Lerner, E. W. D. Luca, and S. Albayrak. Spectral analysis of signed graphs for clustering, prediction and visualization. In *SDM'10*, pages 559–559, 2010.

[16] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In *WWW '10*, 2010.

[17] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Signed networks in social media. In *CHI '10*, 2010.

[18] J. Leskovec, K. Lang, A. Dasgupta, and M. Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123, 2009.

[19] R.-H. Li, J. X. Yu, and J. Liu. Link prediction: the power of maximal entropy random walk. In *CIKM '11*, 2011.

[20] X. Li, X. Su, and M. Wang. Social network-based recommendation: a graph random walk kernel approach. In *JCDL '12*, 2012.

[21] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM '03*, 2003.

[22] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In *KDD '10*, 2010.

[23] T. Lou and J. Tang. Mining structural hole spanners through information diffusion in social networks. In *WWW '13*, 2013.

[24] H. Ma, T. C. Zhou, M. R. Lyu, and I. King. Improving recommender systems by incorporating social contextual information. *ACM Trans. Inf. Syst.*, 29(2):9:1–9:23, Apr. 2011.

[25] M. Makrehchi. Social link recommendation by learning hidden topics. In *RecSys '11*, 2011.

[26] M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.

[27] A. K. Menon and C. Elkan. Link prediction via matrix factorization. In *ECML PKDD'11*, 2011.

[28] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *IMC '07*, 2007.

[29] S. Parthasarathy, Y. Ruan, and V. Satuluri. Community discovery in social networks: Applications, methods and emerging trends. In *Social Network Data Analytics*, pages 79–113. Springer, 2011.

[30] M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. In D. Fensel, K. Sycara, and J. Mylopoulos, editors, *The Semantic Web - ISWC 2003*, volume 2870 of *Lecture Notes in Computer Science*, pages 351–368. Springer Berlin Heidelberg, 2003.

[31] P. Symeonidis, E. Tiakas, and Y. Manolopoulos. Product recommendation and rating prediction based on multi-modal social networks. In *RecSys '11*, 2011.

[32] G. Takács, I. Pilászy, B. Németh, and D. Tikk. Matrix factorization and neighbor based algorithms for the netflix prize problem. In *RecSys '08*, 2008.

[33] J. Tang, T. Lou, and J. Kleinberg. Inferring social ties across heterogenous networks. In *WSDM '12*, 2012.

[34] C. Wang, J. Han, Y. Jia, J. Tang, D. Zhang, Y. Yu, and J. Guo. Mining advisor-advisee relationships from research publication networks. In *KDD '10*, 2010.

[35] R. Xiang, J. Neville, and M. Rogati. Modeling relationship strength in online social networks. In *WWW '10*, 2010.

[36] S.-H. Yang, A. J. Smola, B. Long, H. Zha, and Y. Chang. Friend or frenemy?: predicting signed ties in social networks. In *SIGIR '12*, 2012.

[37] Y. Yang, J. Tang, J. Keomany, Y. Zhao, J. Li, Y. Ding, T. Li, and L. Wang. Mining competitive relationships by learning across heterogeneous networks. In *CIKM '12*, 2012.

[38] J. Ye, H. Cheng, Z. Zhu, and M. Chen. Predicting positive and negative links in signed social networks by transfer learning. In *WWW '13*, 2013.

[39] Z. Yin, M. Gupta, T. Weninger, and J. Han. Linkrec: a unified framework for link recommendation with user attributes and graph structure. In *WWW '10*, 2010.

[40] J. Zhang, C. Wang, Y. Ning, Y. Liu, J. Wang, and P. S. Yu. LaFT-Explorer: Inferring, Visualizing and Predicting How Your Social Network Expands. In *KDD '13*, 2013.

[41] J. Zhang, C. Wang, J. Wang, and P. S. Yu. LaFT-Tree: Perceiving the Expansion Trace of One's Circle of Friends in Online Social Networks. In *WSDM '13*, 2013.

[42] J. Zhang, C. Wang, P. S. Yu, and J. Wang. Learning Latent Friendship Propagation Networks with Interest Awareness for Link Prediction. In *SIGIR '13*, 2013.

[43] J. Zhuang, T. Mei, S. C. Hoi, X.-S. Hua, and S. Li. Modeling social strength in social media community via kernel-based learning. In *MM '11*, 2011.