

國立臺灣大學電機資訊學院資訊網路與多媒體研究所



博士論文

Graduate Institute of Networking and Multimedia
College of Electrical Engineering and Computer Science
National Taiwan University
Doctoral Dissertation

未標記資料之連結發現

Link Discovery with Unlabeled Data

郭宗廷

Tsung-Ting Kuo

指導教授：林守德 博士

Advisor: Shou-De Lin, Ph.D.

中華民國 103 年 1 月

January, 2014

國立臺灣大學博士學位論文

口試委員會審定書

未標記資料之連結發現

Link Discovery with Unlabeled Data

本論文係郭宗廷君（學號 D97944007）在國立臺灣大學資訊網路與多媒體研究所完成之博士學位論文，於民國一百零三年一月六日承下列考試委員審查通過及口試及格，特此證明

口試委員：

林子德

(簽名)

許周華 (指導教授)

張智星

李素琪

張嘉惠

曾新珍

劉文志

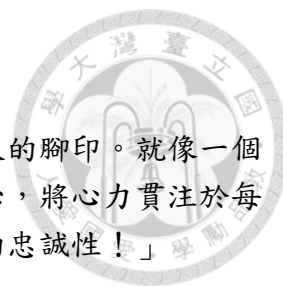
洪宇貝

許永真

逢愛君

所長：

誌謝



「雖然所做的是一樣的行為，但是有遠大的目標，每件事都是偉大的腳印。就像一個王子的童年，和市井小民的童年，絕對不一樣。在日日中數數發心，將心力貫注於每個看似平凡的日子。最偉大的爆發力，就是恆久持續地，對目標的忠誠性！」

五年半的博士生涯，要感謝的人實在太多了！首先當然是指導教授林守德老師，無論在研究教學，論文寫作，競賽活動，做事態度，生活家庭平衡，團隊默契建立，實驗室經營等各方面，都給予最多的支援與指導。感謝博士學位考試予以諸多指導與建議的曾新穆老師，許聞廉老師，許永真老師，洪宗貝老師，張智星老師，張嘉惠老師，李素瑛老師，以及彭文志老師。亦感謝在論文計畫審查階段提供指導的陳信希老師，陳銘憲老師，張智星老師，林軒田老師，以及鄭卜壬老師。此外，也感謝曾指導修習課程的林守德老師，林智仁老師，林軒田老師，李明穗老師，歐陽明老師，及陳彥仰老師。也特別感謝碩士班指導教授曾憲雄老師，以及大學專題指導教授曾新穆老師，在攻讀博士期間，依然給予許多支持和關懷。

感謝駱宏毅，李政德，解巽評，嚴睿，葉蓉蓉，黃安達，沈砥君，林婉真，洪三權，林瑋詩，彭楠贊，林嘉貞，黃宇陽，龔鵬驊，以及曹餘雯等實驗室好伙伴的支援和鼓勵。也感謝陳雅琳小姐，以及網媒所和資工系等，所有職員多年來的幫忙。

十分感謝季松平，孫基康，林俊賢，張烈諄，徐佩玲，王馨佩，黃虹瑜，及宗臣科技和美商國家儀器等，許多事業前輩及伙伴的支援和幫忙。也感謝交大資料所，成大資工系，以及成大慈幼社等，許多好朋友的關懷。

亦非常感謝盧克宙，賴錫源，郭基瑞，林常如，陳靜香，林文清，李妙玲，黃俊昌，林金枝，以及福智文教基金會和里仁公司等，所有長輩及朋友的關懷及支持。也特別感謝臺大福智教職員聯誼會，臺大福智青年校友會，臺大福智青年社，以及福智青年大專班台大校群等，所有教職員生的支持和鼓勵。


當然，家人的支持是很重要的，感謝奶奶，外婆，爸爸，媽媽，姑姑，叔叔，嬸嬸，舅舅，阿姨，妹妹，表弟，表妹，女兒，以及親愛的老婆。

最後，特別感謝如證法師及如英法師的關懷和指導。

謹將本論文，供養最崇敬感恩的師父及上師！

郭宗廷 謹誌
民國一百零三年一月

摘要



許多社群，學術，生物，地理及資訊系統可以用網路來做描述。連結發現是一種在社群網路中確認隱藏連結的研究。然而，某些情況下，針對我們想發現的連結，並無法取得已標記的資料。在此論文中，我們研究一個關於連結發現問題的新面向：發現未標記之連結。我們進一步研究兩個子題，來預測兩種未標記之連結：在異質性網路中未標記之關係連結，以及在同質性網路中未標記之傳播連結。此問題之主要挑戰為缺少標記資料，所以無法直接使用傳統的自動分類方法。為解決此問題，我們設計了以機器學習為基礎的架構，來整合各種不同的資訊，並發現未標記資料的連結。我們也在許多真實世界的資料集上進行實驗，以驗證我們所提出的方法。實驗結果除了顯示我們所提出的方法可以解決此問題，也指出未標記資料之連結發現可以應用在許多不同的實務情境之中。

關鍵字:

連結發現；連結預測；資料探勘；機器學習；社群網路；機率圖形學習模型；自然語言處理

Abstract

Many social, academic, biological, geographical, and information systems can be described by networks. Link discovery is a kind of task aiming at identifying hidden links in a social network. However, in some cases, the labels of the links to be discovered is not available. In this dissertation, we investigate such a novel aspect of the link discovery task: the problem of discovering *unlabeled links*. Specifically, we conduct two studies to predict two kinds of unlabeled links respectively: links that represents unlabeled *relationship* in *heterogeneous* networks, and links that represents unlabeled *diffusion* in *homogeneous* networks. The main challenge of these tasks are the lack of labeled data, thus prevents the direct exploiting of traditional classification approaches. To address this challenge, we design learning-based frameworks to integrate diverse information and solve the corresponding link discovery problems in the two studies. Also, we conduct experiments on various real-world datasets to evaluate our proposed frameworks. The promising experiment results not only demonstrates the usefulness of the proposed models, but also indicates that discovering links without labeled data is feasible in many practical scenarios.

Keywords:

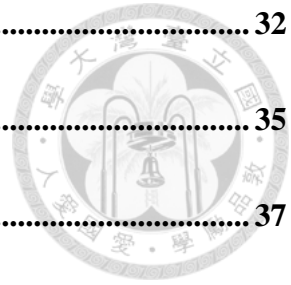
Link discovery; Link prediction; Data mining; Machine learning; Social network; Probabilistic graphical model; Natural language processing

Contents

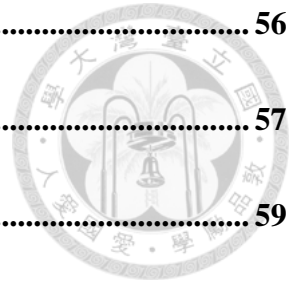


Chapter 1	Introduction	1
1.1	Problem and Motivation	1
1.2	Challenge.....	4
1.3	Methodology, Dataset and Experiment	5
1.4	Literature	7
1.5	Contributions.....	8
1.6	Dissertation Organization.....	9
Chapter 2	Link Prediction Using Aggregative Statistics	10
2.1	Overview	11
2.2	Problem Formulation.....	16
2.3	Methodology	19
2.3.1	Factor Graph Model with Aggregative Statistics (FGM-AS).....	19
2.3.2	An Illustrative Example of FGM-AS.....	22
2.3.3	Attribute-to-Candidate Function.....	24
2.3.4	Candidate-to-Candidate Function	25
2.3.5	Candidate-to-Count Function	26
2.3.6	Ranked-Margin Learning for FGM-AS.....	27
2.4	Experiments	32

2.4.1	Scenarios and Datasets.....	32
2.4.2	Comparing Methods	35
2.4.3	Settings	37
2.4.4	Results	37
2.4.5	Candidate-to-Candidate Verification	39
2.5	Related Work	42
2.5.1	Link Prediction	42
2.5.2	Factor Graph and Max-Margin Learning	43
2.6	Short Summary	45
Chapter 3	Diffusion Prediction of Novel Topics	46
3.1	Overview	47
3.2	The Novel-Topic Diffusion Model	49
3.2.1	The Framework	50
3.2.2	Topic Information.....	51
3.2.3	User Information	52
3.2.4	User-Topic Interaction	53
3.2.5	Global Features.....	54
3.2.6	Complexity Analysis	54
3.3	Experiments	55
3.3.1	Dataset and Evaluation Metric	55



3.3.2	Implementation and Baseline.....	56
3.3.3	Results	57
3.4	Short Summary	59
Chapter 4	Conclusion.....	60
Bibliography	62



List of Algorithms



Algorithm 2-1. Ranked-margin learning algorithm.....	30
Algorithm 2-2. Two-stage inference algorithm.....	31

List of Figures



Figure 2-1. The unseen-type link prediction with aggregative statistics problem in a heterogeneous social network.	13
Figure 2-2. Relational schema of the unseen-type link prediction with aggregative statistics problem shown in Figure 2-1.	18
Figure 2-3. Factor graph model with aggregative statistics (FGM-AS).	19
Figure 2-4. An example of FGM-AS based on Figure 2-1's network.	23
Figure 3-1. The novel-topic diffusion model.	50

List of Tables



Table 1-1. Summary of two studies of link discovery with unlabeled data.	1
Table 1-2. Summary of literatures and our proposed solutions.	7
Table 2-1. Statistics of the datasets.	32
Table 2-2. Mapping of the random variables for the datasets.	34
Table 2-3. Experiment results of our framework (FGM-AS) and all comparing methods (in percentage).	39
Table 2-4. Verification results of candidate-to-candidate functions (in percentage), Pre. = precision, Rec. = recall.	41
Table 3-1. Single-feature results.	57
Table 3-2. Feature combination results.	58



Chapter 1 Introduction

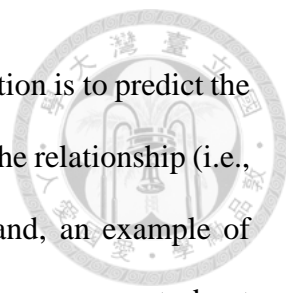
1.1 Problem and Motivation

Many social, academic, biological, geographical, and information systems can be described by networks (tree-structured, homogeneous, heterogeneous, etc.), where nodes represent individuals, and links denote the relations or interactions between nodes [31] [32] [39]. Given such networks, Link discovery tries to estimate the likelihood of the existence of a link between two nodes, based on observed links and the attributes of nodes [16].

Table 1-1. Summary of two studies of link discovery with unlabeled data.

Study	Unlabeled	Network	Publication
Link prediction using aggregative statistics	Relationship	Heterogeneous	[30]
Diffusion prediction of novel topics	Diffusion	Homogeneous	[28]

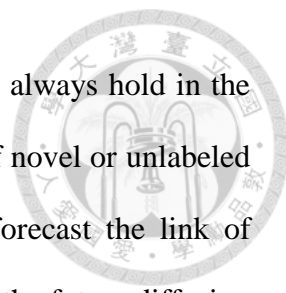
However, in some cases, the links to be discovered is not labeled in training data. Link discovery becomes much more challenging given such scenario. In this dissertation, we investigate the problem of discovering *unlabeled links* (links of specific attributes which are never observed in training data). Specifically, we conduct two studies to predict two kinds of unlabeled links respectively (Table 1-1): links that represent unlabeled *relationship* in *heterogeneous* networks, and links that represent unlabeled *diffusion* in



homogeneous networks. An example of unlabeled relationship prediction is to predict the “like” relationship in Foursquare; due to the privacy policy, labels of the relationship (i.e., whether a user *like* a post or not) is not revealed. On the other hand, an example of unlabeled diffusion prediction is to predict whether a user will response a post about “iPhone 6”, before any post about “iPhone 6” actually exists (that is, we only have labels for posts such as “HTC” and “iPad”, but not for “iPhone 6”). The two unlabeled link prediction studies are described in detail as follows:

(1) **Link prediction using aggregative statistics** (discovering links of unlabeled *relationship* in *heterogeneous* networks) [30]. Most of the social network services allow users to express their opinions (such as “like” or “+1”) to messages posted by other people, and such individual opinions are valuable for many reasons. However, due to privacy concern, opinion holders are sometimes hard to be determined. Fortunately, the aggregative statistics of articles (i.e., how many people like this article) is usually available in such websites. In this study, we target to predict the links of unlabeled relationship. We try to answer a question: can we predict links representing a specific relationship in a heterogeneous network *without* any labeled data, but using the aggregative statistics as well as some attributes provided by the heterogeneous social networks only?

(2) **Diffusion prediction of novel topics** (discovering links of unlabeled *diffusion* in *homogeneous* networks) [28]. Most of the data-driven link discovery approaches assume that in order to train a model and predict the future diffusion of a topic, it is required to obtain historical records about how this topic has propagated in a



homogeneous network. We argue that such assumption does not always hold in the real-world scenario, and being able to forecast the propagation of novel or unlabeled topics is more valuable in practice. In this study, we try to forecast the link of unlabeled diffusion. We try to answer a question: can we predict the future diffusion of *without* labeled training data about how this kind of diffusion has propagated in a homogeneous network?

1.2 Challenge

Although discovering links using unlabeled data is valuable, solving the problems of the two proposed studies it is not trivial because of the following challenges:



(1) **Link prediction using aggregative statistics.** There are three challenges to solve the problem in this study. First, the absence of labeled training data prevents us from performing parameter learning in a straightforward way. Next, in a heterogeneous network, the information of different types of vertices and links are diverse but correlated with each other. A suitable model has to carefully model such correlation together with the aggregative statistics. Finally, since the type is unlabeled, presumably the possible candidate-link count approaches $O(n^2)$ where n is the total number of nodes. When n is large, this can cause serious sparsity problem, while finding the links in such a large space can be very challenging.

(2) **Diffusion prediction of novel topics.** In the problem of this study, the past diffusion behaviors of novel topics are missing, which makes this problem difficult to be solved. That is, without historical training data of the novel topics, it is not easy to maintain reasonable prediction performance.



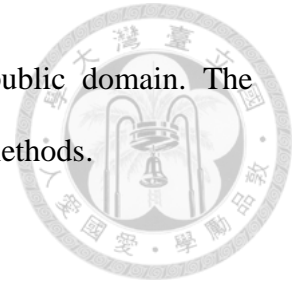
1.3 Methodology, Dataset and Experiment

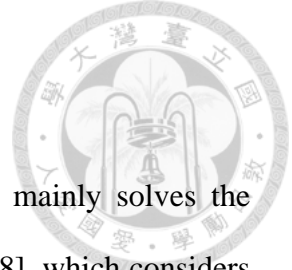
To address the challenges for predicting unlabeled links, we design learning-based frameworks to integrate diverse information and solve the corresponding link discovery problems in the two studies. Also, we conduct experiments on various real-world datasets to evaluate our proposed frameworks and get promising results. The two proposed solutions, datasets, and experiment results, are introduced briefly below.

(1) **Link prediction using aggregative statistics.** In this study, we cannot apply supervised learning methods directly, because we do not have any labeled relationships in the training stage. Instead, we devise a novel unsupervised framework to integrate three kinds of information: candidate, attribute, and count. The proposed framework includes three main components: a three-layer factor graph model and three types of potential functions; a ranked-margin learning algorithm for parameter tuning; and a two-stage inference algorithm for link prediction. Also, we evaluate our method on four diverse scenarios using four datasets: preference prediction (Foursquare), repost prediction (Twitter), response prediction (Plurk), and citation prediction (DBLP). We further exploit nine unsupervised models to solve this problem as baseline, and our approach wins out in all scenarios significantly.

(2) **Diffusion prediction of novel topics.** In this study, we devise a supervised learning framework to solve the problem, because we do have labels for other kinds of diffusions. We exploit the latent semantic information among users, topics, and social connections as features for prediction. Specifically, we integrate four kinds of information: topic, user, user-topic, and global information. Our supervised-learning-

based framework is evaluated on real data collected from public domain. The experiments show promising AUC improvement over baseline methods.





1.4 Literature

As an important task in recent data mining field, link prediction mainly solves the following problems [39]: (1) reconstruction of networks [44] [50] [58], which considers the reconstruction of networks from the observed networks with missing and spurious links; (2) evaluation of network evolving mechanisms [35] [66], which studies the evolving models of networks; and (3) classification of partially labeled networks [14] [65], which is given a network with partial nodes being labeled, predicting the labels of these unlabeled nodes based on the known labels and the network structure.

In terms of methodology, the link prediction approaches can further be divided into two categories: supervised learning [4] [11] [18] [37] [40] [56], and unsupervised learning [1] [3] [6] [17] [22] [24] [43]. However, most of the proposed approaches aim at *seen* links (links of seen node, topic, and type), thus cannot be applied directly to solve the problem of discovering unlabeled links. The literatures and our proposed solutions are summarized in Table 1-2.

Table 1-2. Summary of literatures and our proposed solutions.

	Labeled Data	Unlabeled Data
Unsupervised Learning	[1] [3] [6] [17] [22] [24] [43]	Link prediction using aggregative statistics (Chapter 2)
Supervised Learning	[4] [11] [18] [37] [40] [56]	Diffusion prediction of novel topics (Chapter 3)

1.5 Contributions



The contributions in this dissertation are three-fold:

- **Problem.** We propose a novel problem of discovering unlabeled links, and conduct two related studies to predict links of unlabeled relationship in heterogeneous networks (link prediction using aggregative statistics), and links of unlabeled diffusion in homogeneous networks (diffusion prediction of novel topics).
- **Solution.** We devise two diverse learning-based frameworks, to integrate the diverse information and solve the unlabeled link discovery problems. For the link prediction using aggregative statistics task, we integrate candidate, attribute and count information in an unsupervised learning framework. For the diffusion prediction of novel topics task, we integrate the topic, user, user-topic, and global information in a supervised learning framework.
- **Experiment.** We conduct experiments on real-world datasets (Foursquare, Twitter, Plurk, and DBLP). The results show that our proposed frameworks provide reasonably high performance and can solve the unlabeled link prediction problems.

1.6 Dissertation Organization

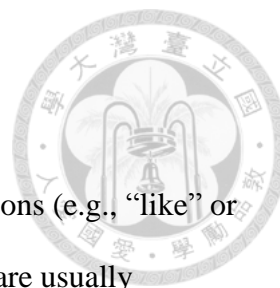
The remainder of this dissertation is organized as follows. In the next chapter, we present the link prediction using aggregative statistics problem and explain how we tackle this problem. In Chapter 3, we introduce and solve the diffusion prediction of novel topics problem. Then, in Chapter 4, we provide concluding remarks of this dissertation.



Chapter 2 Link Prediction Using Aggregative Statistics



The concern of privacy has become an important issue for online social networks. In services such as Foursquare.com, whether a person likes an article is considered private and therefore not disclosed; only the aggregative statistics of articles (i.e., how many people like this article) is revealed. This study tries to answer a question: can we predict the opinion holder in a heterogeneous social network without any labeled data? This question can be generalized to a *link prediction with aggregative statistics* problem. This study devises a novel unsupervised framework to solve this problem, including two main components: (1) a three-layer factor graph model and three types of potential functions; (2) a ranked-margin learning and inference algorithm. Finally, we evaluate our method on four diverse prediction scenarios using four datasets: preference (Foursquare), repost (Twitter), response (Plurk), and citation (DBLP). We further exploit nine unsupervised models to solve this problem as baselines. Our approach not only wins out in all scenarios, but on the average achieves 9.79% AUC and 12.81% NDCG improvement over the best competitors.



2.1 Overview

Most of the social network services allow users to express their opinions (e.g., “like” or “+1”) to messages posted by other people. Such individual opinions are usually valuable: companies can identify a specific customer’s preference, and government can recognize the will or desire of target influential person.

However, due to privacy concern, opinion holders are sometimes concealed. An example is Foursquare.com, a popular location-based social network websites. In Foursquare, users can post tips to certain venues of their interest, and other people may “like” the tips. Nevertheless, the information about which user likes which tip is generally not available to public due to the privacy concern.

Another example is Pinterest.com, which is a pin-board-style photo sharing website. In Pinterest, users can “like” or “*repin*” others’ images, but only a little portion of such information is available due to internal limitation of Pinterest (only first 24 “like” and first 8 “*repin*” are shown on the webpage). Thus, it is difficult to gather a full spectrum of information about each individual’s opinion under such circumstances.

Fortunately, *aggregative statistics* of opinions are usually available. For example, the total count of “like” of each tip in Foursquare is accessible, and the total count of “like” and “*repin*” of an image in Pinterest is also obtainable. Such aggregative statistics are important because it is usually the only available clue to understand the quality of certain item without violating the policy rule. Hence, this study tries to address a problem: can we predict a link between a user and an item (e.g., whether a user likes a tip) using the

aggregative statistics together with other information in a heterogeneous social network?



We generalize the question to an *unseen-type link prediction with aggregative statistics* problem. The term *unseen* is used because we assume it is not possible to obtain which person likes which tip from data (therefore, such “like” link can be regarded as a kind of relationship that is previously unseen). From link prediction point of view, one can assume there is *no* labeled training data available of such type of links.

An example we use through this study is a network gathered from Foursquare (Figure 2-1). There are 7 nodes and 7 links with 3 node types (users, items, and categories) and 3 link types (be-friend-of, own, and belong-to). We want to predict the existence of “like” links (e.g., whether user u_2 likes item r_2 or not) using the aggregative statistics (e.g., total like count of the item r_2 is $t(r_2) = 1$). Note that the links of “like” type is unseen, which means we do not see such link at all in the data.

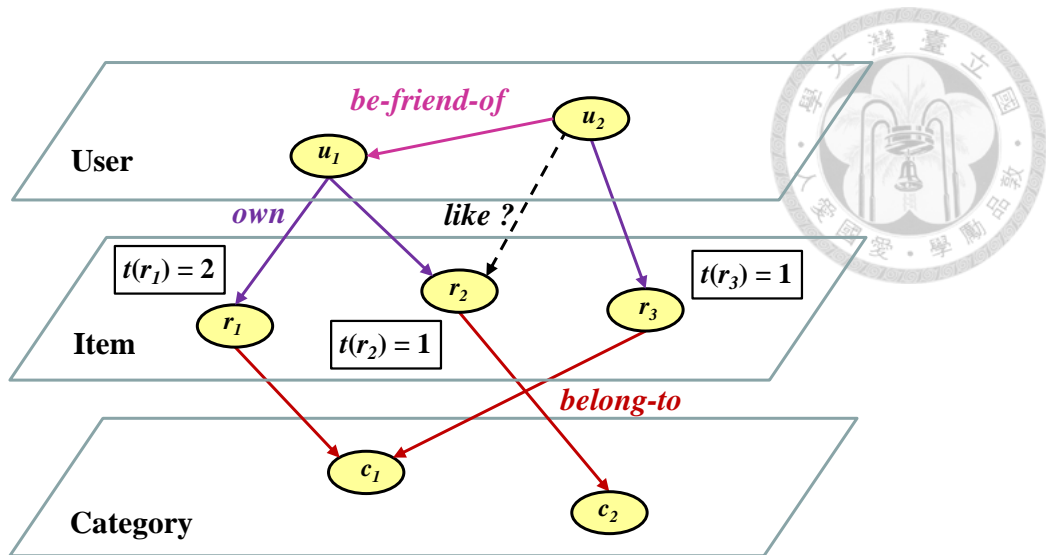


Figure 2-1. The unseen-type link prediction with aggregative statistics problem in a heterogeneous social network.

Most of the link prediction literatures aim at predicting links of *seen* types (i.e., some labeled historical links are available as the training data) [35] [39] [62], thus cannot be applied to our problem. Some researchers predict links of unseen types using external node group information [33], but those information are not always available. As in the Foursquare example, the only available information in our problem is the aggregative statistics. Nevertheless, our problem is non-trivial due to the following three challenges:

- **Lack of labeled data.** The absence of labeled training data prevents us from performing parameter learning in a straightforward way.
- **Diverse information.** In a heterogeneous social network, the information of different types of nodes and links are diverse but correlated with each other. A suitable model is needed to represent such correlation with aggregative statistics.

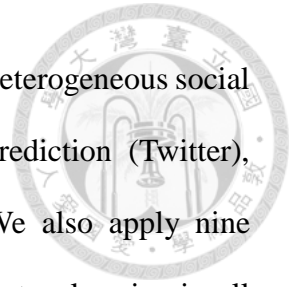


- **Sparsity of links.** Since the type is unseen, presumably the possible candidate-link count approaches $O(n^2)$ where n is the total number of nodes. When n is large, this can cause serious sparsity problem, while finding the links in such a large space can be very challenging.

In this study, we try to address these challenges by proposing a novel unsupervised probabilistic graphical model. First, we devise a factor graph model with three layers of random variables (candidate, attribute, and count) to infer the existence of unseen-type links. Second, we define three types of potential functions (attribute-to-candidate, candidate-to-candidate, and candidate-to-count) to integrate diverse information into the factor graph model. Third, we design a ranked-margin learning algorithm to automatically tune the parameters using aggregative statistics. Finally, we design a two-stage inference algorithm to update the candidate-to-count potential functions, and optimize the outputs. The main contributions of this study are as below:

- We propose and formulate a novel yet practical problem to predict the links of unseen-type using aggregative statistics in heterogeneous social networks.
- We devise an unsupervised learning framework to solve the above-mentioned problem. Note that the framework we propose can be exploited not only for probabilistic graphical models, but for all kinds of general situations where only aggregative statistics are available for learning.

- We evaluate our method on four diverse scenarios using different heterogeneous social network datasets: preference prediction (Foursquare), repost prediction (Twitter), response prediction (Plurk), and citation prediction (DBLP). We also apply nine unsupervised models for this problem as baseline. Our model not only wins in all scenarios, but also achieves on the average 9.79% AUC and 12.81% NDCG improvement over the best comparing methods.





2.2 Problem Formulation

We start by formulating the problem.

Definition 1. **Heterogeneous social network** $N = (V, E, \Omega_V, \Omega_E)$ is a directed graph, where V is a set of nodes, Ω_V is a set of node labels, Ω_E is a set of link labels, and $E \subseteq V \times \Omega_E \times V$ is a set of links.

The function $type(v) \rightarrow l_V$ maps node v onto its node label $l_V \in \Omega_V$. Similarly, given a triplet $\langle source, link-label, target \rangle$ as a link, the function $type(e) \rightarrow l_E$ maps link e onto its link label $l_E \in \Omega_E$.

For the example shown in Figure 2-1, there are 7 nodes and 7 links, with $\Omega_V = \{ "user", "item", "category" \}$ and $\Omega_E = \{ "be-friend-of", "own", "belong-to" \}$. For brevity, we denote $U \subseteq V$ as the set of node for type = "user", $R \subseteq V$ for type = "item", and $C \subseteq V$ for type = "category".

The relationship between node labels and link labels can be enumerated. For instance, a user u may "be-friend-of" another user v (i.e., $\langle u, "be-friend-of", v \rangle$); a user u may "own" an item r (i.e., $\langle u, "own", r \rangle$), and an item r may "belong-to" a category c (i.e., $\langle r, "belong-to", c \rangle$).

It should be noted that the number of items, $|R|$, is equivalent to the total number of "own" links, and is also equivalent to the total number of "belong-to" links (i.e., each item can only be owned by one user, and can only belong to one category).



Definition 2. **Unseen-type links** is a set of links with a special type “?”; links of such type do not appear in a given heterogeneous social network. That is, unseen-type links $\Phi = \{ \varphi \mid \varphi = \langle source, “?”, target \rangle, type(source) \in \Omega_V, type(target) \in \Omega_V, “?” \notin \Omega_E \}$.

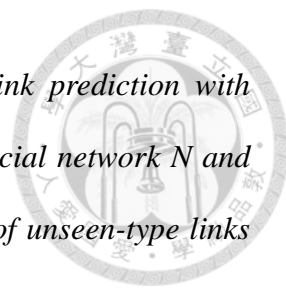
For the example in Figure 2-1, the unseen-type links denote the “like” behavior. That is, $\Phi = \{ \langle u, “like”, r \rangle \}$ denotes the set of links that user u likes item r . We use $\langle u, r \rangle$ to denote the candidate pairs of unseen-type links, and there are $|U| \cdot |R| = 6$ plausible candidate pairs in Figure 2-1.

Definition 3. **Aggregative statistic** is the total unseen-type link count of a target node. In other words, the aggregative statistic of a node $v \in V$ is $\sigma(v, \Phi) = | \{ \varphi \mid \varphi = \langle source, “?”, target \rangle \in \Phi, target = v \} |$, which is a non-negative integer.

In our example, the aggregative statistic of an item $r_2 \in R$ is $\sigma(r_2, \Phi) = | \{ \varphi \mid \varphi = \langle u, “like”, r \rangle \in \Phi, r = r_2 \} | = 1$.

Definition 4. **Aggregative statistics of a heterogeneous social network** $T(N, \Phi) = \{ \langle v, \sigma(v, \Phi) \rangle \mid v \in V \}$ is the set of aggregative statistics of the unseen links for a heterogeneous social network N .

In Figure 2-1, the aggregative statistics of heterogeneous social network N is $T(N, \Phi) = \{ \langle r_1, 2 \rangle, \langle r_2, 1 \rangle, \langle r_3, 1 \rangle \}$.



Based on above definitions, we formulate the *unseen-type link prediction with aggregative statistics* problem as follows: given a heterogeneous social network N and corresponding aggregative statistics $T(N, \Phi)$, predict the existence of unseen-type links Φ .

The relational schema for our example is shown in Figure 2-2: given the heterogeneous social network (3 types of nodes and 3 types of edges) and aggregative statistics of “like”, predict whether each $\langle u, \text{“like”}, r \rangle$ exists or not, where $u \in U$ and $r \in R$.

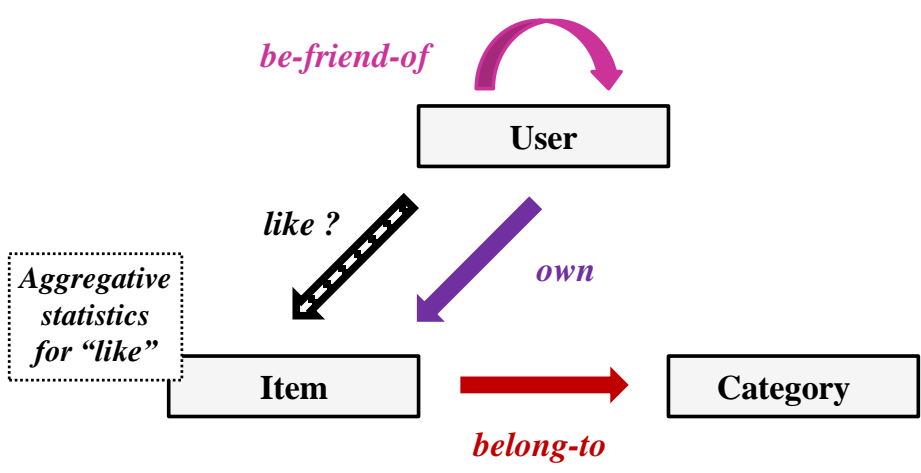


Figure 2-2. Relational schema of the unseen-type link prediction with aggregative statistics problem shown in Figure 2-1.



2.3 Methodology

We first propose to solve this problem using a probabilistic model. Then, we use an illustrative example to demonstrate our model. Finally, we describe a novel learning algorithm utilizing the aggregative statistics to learn the model parameters, as well as a two-stage inference algorithm to predict unseen-type links.

2.3.1 Factor Graph Model with Aggregative Statistics (FGM-AS)

To handle this problem, we propose a novel probabilistic graphical model: *factor graph model with aggregative statistics* (FGM-AS), as shown in Figure 2-3. There are three layers of variables in FGM-AS:

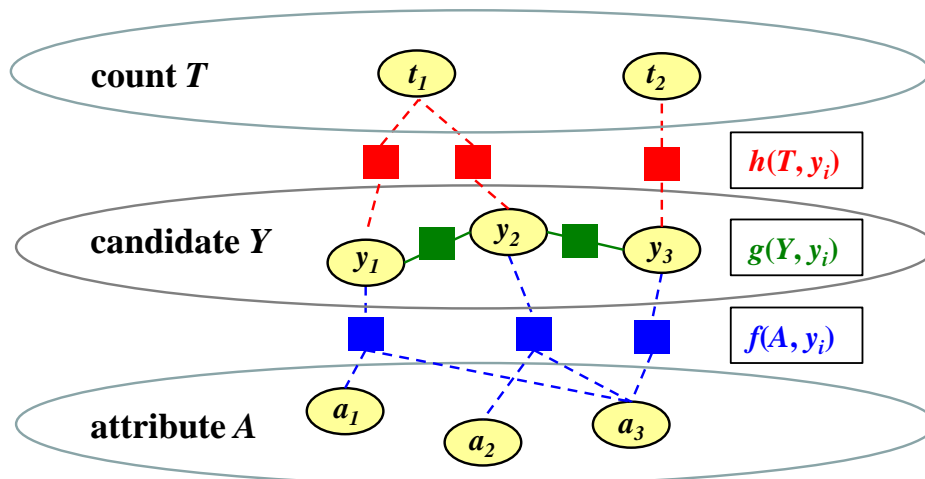


Figure 2-3. Factor graph model with aggregative statistics (FGM-AS).

- **Candidate:** the binary random variables Y in the *candidate* layer represent all unseen-type links to be predicted. They either exist (positive) or not exist (negative). Each



candidate y_i can be regarded as a pair of user and item, $\langle u, r \rangle$. Also note that some y 's might point to the same users while some might share the same item.

- **Attribute:** the random variables A in the *attribute* layer carry attribute information (e.g., a_1 represents the degree of the source node and a_2 represents the degree of the target node) of the candidate links.
- **Count:** the random variables T in the *count* layer encode the aggregative statistics of the items. Note that t is a one-to-one mapping of an item r , but a one-to-many mapping of y because there are some y 's sharing the same item (e.g., candidate y_1 and y_2 point to the same t_1 as they have the same item r).

Together with the random variables, we also propose three types of potential functions:

- **Attribute-to-candidate functions:** we define this type of potential function as a linear exponential function

$$f(A, y_i) = \frac{1}{Z_\alpha} \exp\{\alpha \cdot f'(A, y_i)\} \quad (1)$$

where $f'(A, y_i)$ is a vector of functions representing the associations between a candidate and its attributes (see Section 2.3.3 for a detailed example), α is a vector of the corresponding weights, and Z_α is a normalization factor. Note that each candidate y can connect to multiple attributes.



- **Candidate-to-candidate functions:** this type of potential function is defined as

$$g(Y, y_i) = \frac{1}{Z_\beta} \exp\{\beta \cdot g'(Y, y_i)\} \quad (2)$$

where $g'(Y, y_i)$ is a vector of functions representing the relationships between candidate random variables (see Section 2.3.4 for a detailed example), β is a vector of weights, and Z_β is a normalization factor.

- **Candidate-to-count functions:** this type of potential function is defined as

$$h(T, y_i) = \frac{1}{Z_\gamma} \exp\{\gamma \cdot h'(T, y_i)\} \quad (3)$$

where $h'(T, y_i)$ is a vector of functions representing the constraints of aggregative statistics (see Section 2.3.5 for a detailed example), γ is a vector of weights, and Z_γ is a normalization factor. More precisely, this type of potential functions adhere to the condition: the sum of predicted marginal probability of the candidate random variables of each item should be as close to the total count of that item as possible.

According to the FGM-AS model, when the candidates, attributes and counts are known, we can define the joint distribution as

$$P(A, T, Y) = \prod_i f(A, y_i) \cdot g(Y, y_i) \cdot h(T, y_i) \quad (4)$$

Therefore, the marginal probability of candidate random variable y_i being positive (e.g., *like*) is

$$P(A, T, Y, y_i) = \sum_j P(A, T, Y, y_j), y_j \in Y / \{y_i\} \quad (5)$$



The marginal probability $P(A, T, Y, y_i = 1)$ is the desired output in our problem, as it tells us for $y_i = \langle u, r \rangle$, how likely u likes r .

2.3.2 An Illustrative Example of FGM-AS

We believe that FGM-AS is a general graphical model for solving the unseen-type links prediction problem. The three layers of random variables and the three types of potential functions can be flexibly defined for different application context. Here we use FGM-AS to predict whether a user likes an item or not. Figure 2-4 illustrates an example of FGM-AS, which is built from the heterogeneous social network shown in Figure 2-1. The three layers of random variables are defined as:

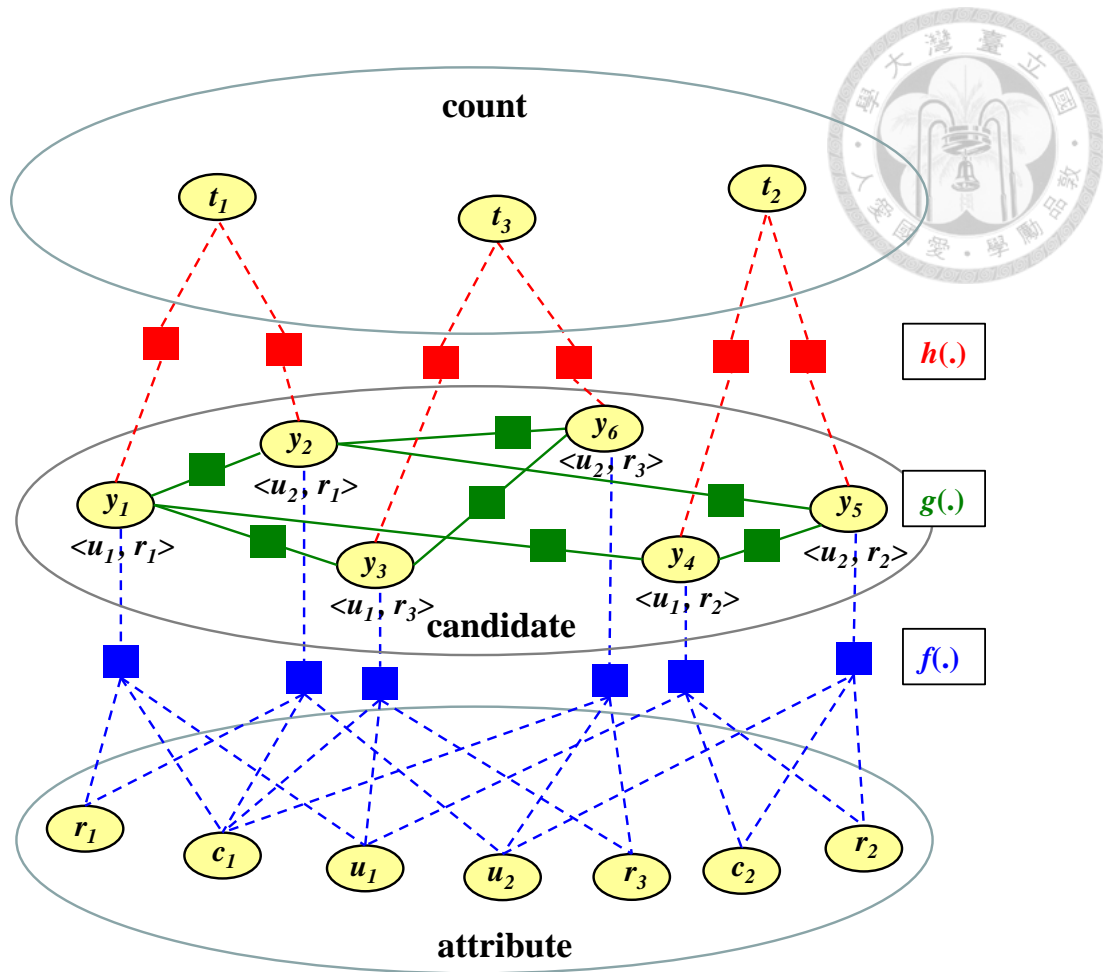
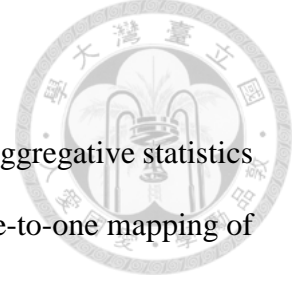


Figure 2-4. An example of FGM-AS based on Figure 2-1's network.

- **Candidate:** candidate random variables $Y = \{ y_i \mid i = 1, 2, \dots, |U| \cdot |R| \}$ represent the set of plausible links $\langle u, r \rangle$ to be predicted. In other words, each pair $y_i = \langle u, r \rangle$ indicates whether the user u likes the item r . For example, $y_1 = \langle u_1, r_1 \rangle$ represents whether user u_1 likes item r_1 . Note that u_1 is not necessarily the owner of r_1 .
- **Attribute:** attribute random variables $A = U \cup R \cup C$ contain three groups of information: users $U = \{ u_1, u_2, \dots, u_{|U|} \}$, items $R = \{ r_1, r_2, \dots, r_{|R|} \}$, and categories $C = \{ c_1, c_2, \dots, c_{|C|} \}$. We use $u(y_i)$ to denote the corresponding user, $r(y_i)$ to denote the corresponding item, and $c(y_i)$ to denote the corresponding category of y_i .



- **Count:** count random variables $T = \{t_1, t_2, \dots, t_{|R|}\}$ represent the aggregative statistics (total like count) of each item. Note that $|T| = |R|$ because t is a one-to-one mapping of r . We use $t(y_i)$ to denote the corresponding count of y_i .

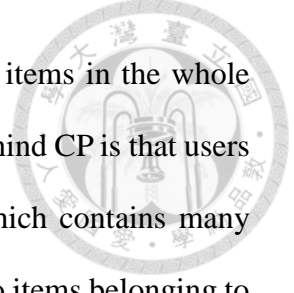
The design of the three potential functions is described in the following three subsections.

2.3.3 Attribute-to-Candidate Function

According to Equation (1), we define $f'(A, y_i) = \langle f_{UF}(u(y_i)), f_{IO}(u(y_i), r(y_i)), f_{CP}(c(y_i)) \rangle$.

The functions f_{UF} , f_{IO} and f_{CP} are based on user friendship, item ownership, and category popularity, which are defined below:

- **User friendship (UF) function:** $f_{UF}(u(y_i)) =$ the number of friends of $u(y_i)$. The intuition behind UF is that we believe the number of friends of a user can influence his / her tendency to like an item. In Figure 2-1, $f_{UF}(u(y_1)) = f_{UF}(u_1) = 1$, because user u_1 has only one friend (which is u_2).
- **Item ownership (IO) function:** $f_{IO}(u(y_i), r(y_i)) = 1$ if $r(y_i)$ is owned by $u(y_i)$, otherwise 0. The intuition behind IO is that we believe whether a user likes an item or not depends significantly on whether this item is owned by this user. In Figure 2-1, $f_{IO}(u(y_1), r(y_1)) = f_{IO}(u_1, r_1) = 1$, because u_1 owns r_1 .



- **Category popularity (CP) function:** $f_{CP}(c(y_i))$ = the number of items in the whole dataset that belongs to the same category as $c(y_i)$. The intuition behind CP is that users tend to like items belonging to a hot category (i.e., category which contains many items). In Figure 2-1, $f_{CP}(c(y_1)) = f_{CP}(c_1) = 2$, because there are two items belonging to c_1 .

2.3.4 Candidate-to-Candidate Function

According to Equation (2), we define $g'(Y, y_i) = \langle \sum_j g_{OI}(y_i, y_j), \sum_j g_{FI}(y_i, y_j), \sum_j g_{OF}(y_i, y_j), \sum_j g_{CC}(y_i, y_j), \sum_j g_{CI}(y_i, y_j) \rangle, y_j \in Y / \{y_i\}$. The functions g_{OI} , g_{FI} , g_{OF} , g_{CC} and g_{CI} are based on owner, friend, owner-friend, co-category, and common-interest relationships, which are defined as follows:

- **Owner-identification (OI) function:** $g_{OI}(y_i, y_j) = 1$ if $\langle u(y_i), \text{"own"}, r(y_i) \rangle \in E, \langle u(y_j), \text{"own"}, r(y_j) \rangle \in E$, and $u(y_i) = u(y_j)$; otherwise 0. The intuition is that an owner tends to like all his / her items. For example in Figure 2-1, u_1 likes both r_1 and r_2 , because u_1 owns both items. Therefore, there will be a relation between y_1 and y_4 in Figure 2-4.
- **Friend-identification (FI) function:** $g_{FI}(y_i, y_j) = 1$ if $\langle v, \text{"own"}, r(y_i) \rangle \in E, \langle v, \text{"own"}, r(y_j) \rangle \in E, u(y_i) = u(y_j)$, and $v \in \text{friend}(u(y_i))$; otherwise 0. The intuition is that a person may like friend's items. For example, u_2 likes both r_1 and r_2 , because u_2 's friend u_1 owns both items. Therefore, there will be a relation between y_2 and y_5 .

- **Owner-friend (OF) function:** $g_{OF}(y_i, y_j) = 1$ if $\langle u(y_i), \text{"own"}, r(y_i) \rangle \in E$, $r(y_i) = r(y_j)$, and $u(y_i) \in \text{friend}(u(y_j))$; otherwise 0. The intuition is that if an owner likes his / her own item, his / her friends tend to like the item too. For example, if u_1 likes his / her item r_1 , then his / her friend u_2 tends to like r_1 as well. In other words, there will be a relation between y_1 and y_2 .
- **Co-category (CC) function:** $g_{CC}(y_i, y_j) = 1$ if $\langle u(y_i), \text{"own"}, r(y_i) \rangle \in E$, $u(y_i) = u(y_j)$, and $c(y_i) = c(y_j)$; otherwise 0. The intuition is: the extent an owner likes the item will be similar to the extent of the owner likes other items in the same category. For example, if u_1 tends to like item r_1 , then u_1 may also like r_3 , because r_1 and r_3 are in the same category c_1 . Thus, there is a relation between y_1 and y_3 .
- **Common-Interest (CI) function:** $g_{CI}(y_i, y_j) = 1$ if $\langle u(y_i), \text{"be-friend-of"}, u(y_j) \rangle \in E$, and $r(y_i) = r(y_j)$; otherwise 0. The intuition is that if a user likes an item, his / her friends tend to like the item too. For example, if u_1 likes an item r_2 , then his / her friend u_2 tends to like r_2 as well. In other words, there will be a relation between y_4 and y_5 .

2.3.5 Candidate-to-Count Function

According to Equation (3), we define $h'(T, y_i) = \langle h_{CT}(y_i, t(y_i)) \rangle$. The function h_{CT} is defined as:

$$h_{CT}(y_i, t(y_i)) = 1 - \left| \frac{t(y_i) - \sum_{y_j \in Y, r(y_j) = r(y_i)} P(A, T, Y, y_j = 1)}{|U|} \right| \quad (6)$$

The summation term in Equation (6) sums up all the probabilities of a certain item $r(y_i)$ being liked by each user, which we hope to be as close to the observed “like” count of this item as possible. Thus, the difference of this term and $t(y_i)$ represents how close the prediction to the known aggregative statistics is. We divide this difference by $|U|$ for normalization purpose. Ideally, the difference is 0, and thus $h_{CT}(y_i, t(y_i)) = 1$. Also, $0 \leq h_{CT}(y_i, t(y_i)) \leq 1$.

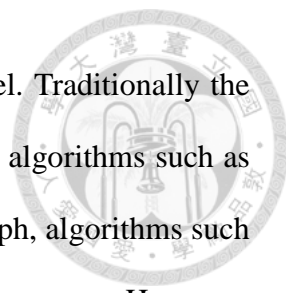
It should be noted that $P(A, T, Y, y_j = 1)$ are not random variables anymore but the posterior probability of them. Therefore, the conventional exact or approximated inference methods cannot be applied directly. To update accordingly, we design a two-stage inference algorithm, which is described at the end of Section 2.3.6.

2.3.6 Ranked-Margin Learning for FGM-AS

The key factor that contributes to the success of FGM-AS lies in the algorithm’s capability of learning the parameters without labeled data. Here we discuss the main idea. Given a parameter configuration $\theta = (\alpha, \beta, \gamma)$ and based on Equation (1) – (4), the joint probability $P(A, T, Y)$ can be written as

$$\begin{aligned} P(A, T, Y) &= \frac{1}{Z} \prod_i \exp\{\theta \cdot (f'(A, y_i), g'(Y, y_i), h'(T, y_i))\} \\ &= \frac{1}{Z} \exp\{\theta \cdot \sum_i s(y_i)\} = \frac{1}{Z} \exp\{\theta \cdot S\} \end{aligned} \quad (7)$$

where all potential functions for a y_i is written as $s(y_i) = \langle f'(A, y_i), g'(Y, y_i), h'(T, y_i) \rangle$, $Z = Z_\alpha Z_\beta Z_\gamma$, and $S = \sum_i s(y_i)$.



Now, we will discuss how to learn the parameters of the model. Traditionally the idea of *maximum-likelihood estimation* (MLE) can be exploited and algorithms such as EM can be applied to achieve this goal. Alternatively for a factor graph, algorithms such as gradient decent can be exploited to greedily search in the parameter space. However, in our scenario, the absence of labels eliminates the possibility of exploiting MLE strategy for learning. Moreover, even if one can somehow come up with certain approximated objective to be maximized in the M-step of EM, the total number of hidden variables in this graph grows to $|U| \cdot |R|$, which can lead to very high computational cost for parameter learning.

To effectively and efficiently perform the learning task, we propose a novel idea to maximize the *ranked-margin* of the instances, incorporating the aggregative statistics into the objective function. The intuition is to assume the count for an item $r(y_i)$ is $t(y_i)$, which means that among all candidate users, only $t(y_i)$ of them like this object.

Therefore, during learning we want to adjust the parameter so that the top $t(y_i)$ users have very high probabilities of liking this item while the rest have very low probabilities of liking it. To realize this idea, we propose to do the following. For each item r , first rank each user u_i based on the marginal probability of $y = \langle u_i, r \rangle$. Then, let $P(Y_r^{upper})$ be the average positive marginal probabilities for the top $t(y_i)^{\text{th}}$ candidate pairs, and $P(Y_r^{lower})$ be the average marginal probabilities for the rest of the candidate pairs, for all y_i of which $r(y_i) = r$. Finally, given $t(y_i)$, we want to adjust the parameters to maximize

$$Diff(Y_r^{margin}) = P(Y_r^{upper}) - P(Y_r^{lower}) \quad (8)$$

An extreme example is that the marginal probability of the top $t(y_i)$ candidate pairs are all 1, while the rest are all 0. In this case $Diff(Y_r^{margin}) = 1 - 0 = 1$. Another extreme example is that the marginal probability of all candidate pairs are equal, which results in $Diff(Y_r^{margin}) = 0$. Thus, $0 \leq Diff(Y_r^{margin}) \leq 1$.

Based on the above idea and Equation (8), we define the log-likelihood objective function to be maximized as

$$\begin{aligned}
 O(\theta, r) &= \log P(Y_r^{margin}) = \log \sum_{Y_r^{margin}} \frac{1}{Z} \exp\{\theta \cdot S\} \\
 &= \log \sum_{Y_r^{upper}} \exp\{\theta \cdot S\} - \log \sum_{Y_r^{lower}} \exp\{\theta \cdot S\}
 \end{aligned} \tag{9}$$

Besides the intuitiveness of Equation (8) with respect to the count as mentioned, there are two other advantages of using Equation (9) as our objective function. First, it should be noted that computing the normalization factor Z in Equation (7) is very time-consuming. However, for Equation (9), we can essentially eliminate Z to avoid the high computational cost during learning. Second, the gradient of Equation (9) can be obtained through sampling using any inference algorithm (as shown below).

To maximize the objective function, we exploit an idea similar to the Stochastic Gradient Descent (SGD) method, as shown in Algorithm 1. We calculate the gradient and update the parameters for each item iteratively until convergence, then move on to the next item (η is the learning rate of our algorithm). The gradient for each parameter θ and item r is

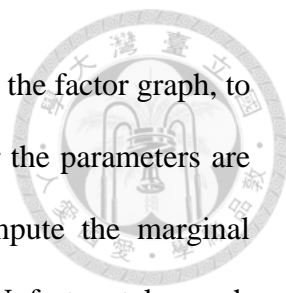


$$\begin{aligned}
\frac{\partial(\theta, r)}{\partial \theta} &= \frac{\partial \left(\log \sum_{Y_r^{upper}} \exp\{\theta \cdot S\} - \log \sum_{Y_r^{lower}} \exp\{\theta \cdot S\} \right)}{\partial \theta} \\
&= \frac{\sum_{Y_r^{upper}} \exp\{\theta \cdot S\} \cdot S}{\sum_{Y_r^{upper}} \exp\{\theta \cdot S\}} - \frac{\sum_{Y_r^{lower}} \exp\{\theta \cdot S\} \cdot S}{\sum_{Y_r^{lower}} \exp\{\theta \cdot S\}} \\
&= \mathbb{E}_{P_{\theta}(Y_r^{upper})} S - \mathbb{E}_{P_{\theta}(Y_r^{lower})} S
\end{aligned} \tag{10}$$

where $\mathbb{E}_{P_{\theta}(Y_r^{upper})} S$ and $\mathbb{E}_{P_{\theta}(Y_r^{lower})} S$ are two expected values of S . The expected values can be obtained naturally using approximated inference algorithms, such as Gibbs Sampling or Contrastive Divergence. It should be noted that the proposed ranked-margin algorithm can be exploited not just for graphical model, but also for other learning models as long as the gradient of the expected difference can be calculated.

Input: FGM-AS, learning rate η
Output: $P(A, T, Y, y_i = 1)$ for all $y_i \in Y$
Initialize all elements in parameter configuration $\theta = 1$
repeat
 Run inference method using current θ to obtain $P(A, T, Y, y_i = 1)$
 Compute potential function values S according to Eq. (1) – (7)
 foreach $r \in R$ **do**
 Compute gradient $\frac{\partial O(\theta, r)}{\partial \theta}$ using S according to Eq. (10)
 $\theta = \theta + \eta \cdot \frac{\partial O(\theta, r)}{\partial \theta}$
 end
until *convergence*

Algorithm 2-1. Ranked-margin learning algorithm.



In Algorithm 2-1, we need to perform an inference algorithm on the factor graph, to obtain the marginal probability of each candidate pair y . Also, after the parameters are learned, we need to apply the inference algorithm again to compute the marginal probability, representing how likely the person likes the item. Unfortunately, such inference cannot directly be done as $P(A, T, Y, y_i = 1)$ in Equation (6) requires the posterior probabilities of y .

Thus, we design a two-stage inference algorithm (Algorithm 2-2). In the first stage, we perform general inference method using $f(A, y_i)$ and $g(Y, y_i)$ only (by assigning all $h(T, y_i) = 1$) to initialize $P(A, T, Y, y_i = 1)$. In the second stage, we compute $h(T, y_i)$ using $P(A, T, Y, y_i = 1)$, and then perform inference one more time. This way, we integrate the posterior information into the inference process.

Input: FGM-AS, parameter configuration θ

Output: $P(A, T, Y, y_i = 1)$ for all $y_i \in Y$

Initialize all $y_i = 0$, all $h(T, y_i) = 1$

stage 1

Calculate $f(A, y_i)$ and $g(Y, y_i)$ according to Eq. (1), (2)

Run an inference method using θ to obtain $P(A, T, Y, y_i = 1)$

stage 2

Calculate $h(T, y_i)$ using $P(A, T, Y, y_i = 1)$ according to Eq. (3), (6)

Run an inference method using θ to obtain final $P(A, T, Y, y_i = 1)$

Algorithm 2-2. Two-stage inference algorithm.



2.4 Experiments

Here we want to verify the generalization of our model by testing whether it can be applied to datasets in four different scenarios. We also want to verify the usefulness of the potential functions.

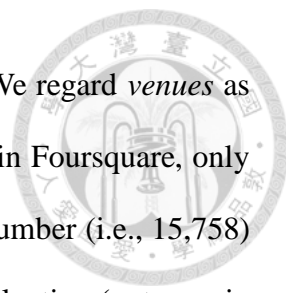
2.4.1 Scenarios and Datasets

We study the following four types of scenarios of the unseen-type link prediction problem, each with a real-world dataset. The statistics of the datasets are shown in Table 2-1.

Table 2-1. Statistics of the datasets.

Property		Foursquare	Twitter	Plurk	DBLP
Node	User	71,634	69,026	190,853	102,304
	Item	180,684	55,375	352,376	221,935
	Category	16,961	100	100	100
	Total	269,279	124,501	543,329	324,339
Link	Be-friend-of	724,378	21,979,021	2,151,351	245,391
	Own	180,684	55,375	352,376	221,935
	Belong-to	180,684	55,375	352,376	221,935
	Unseen	15,758	79,918	804,404	123,479
	Total	1,101,504	22,169,689	3,660,507	812,740

- **Preference prediction.** In location-based social network services, we are interested in predicting whether users like a tip at a venue (i.e., add the tip into their like list). We extract the social network website *Foursquare* as the dataset for evaluation and consider *like* as the unseen-type link. We select all venues located in New York, collect



all tips for these venues, and identify users who posted the tips. We regard *venues* as categories, and *tips* as items. Note that due to the privacy policy in Foursquare, only the total like count of each tip is revealed. There is very limited number (i.e., 15,758) of unseen-type links revealed, which become ground truth for evaluation (not seen in training).

- **Repost prediction.** In social network websites, we are interested in predicting whether users will re-blog or retweet a post. Therefore, we use *Twitter* as the dataset, which is collected from [15]. Twitter is one of the most famous micro-blog website, and has been used to verify several models with different purposes [15] [20] [47]. In this study, we consider *retweet* as the unseen-type link. We keep users who have two or more friends, and have tweeted or retweeted more than once. Then, we perform stemming to identify 100 most popular *terms* in tweets as categories while each *tweet* is regarded as an item. For example, if a user v posts a tweet r , and later another user u retweets this tweet (with the “RT@” keyword), we consider an unseen-type link exists from u to r .
- **Response prediction.** In micro-blog services, we are interested in predicting whether users will respond to a post. We use *Plurk* dataset in this scenario. Plurk is a popular micro-blog service in Asia with more than 5 million users, and has been used in studies of diffusion prediction [28], diffusion model evaluation [27], and mood classification [7]. This dataset is collected from 01/2011 to 05/2011. In this study, we consider *response-to-message* as the unseen-type link. We manually identify the 100 most popular *topics* as categories, and regard *messages* as items. For example, if a person v



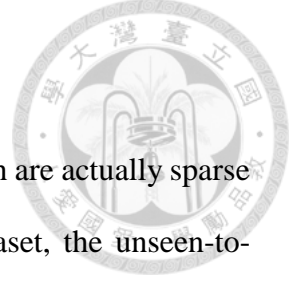
posts a message r , and later another person u responds to this message, we consider an unseen-type link exists from u to r .

- **Citation prediction.** In academic indexing and searching services, we are interested in predicting whether researchers will cite a paper. Therefore, we use *DBLP* [34] dataset collected from ArnetMiner [52], version 5. In this study, we consider *citation-to-paper* as the unseen-type link. We first perform stemming, and then identify the 100 most popular *terms-in-titles* as categories, and regard *papers* as items. For example, if a researcher v published a paper r , and later another researcher u cites r , we consider an unseen-type link exists from u to r . Also, we consider two researchers as friend if they have been co-authors of at least one paper in the past.

The mapping of the information in the four abovementioned datasets to the random variables in FGM-AS is shown in Table 2-2. Note that in the above four datasets (Foursquare, Twitter, Plurk, and DBLP), we hide all unseen-link information as ground truth to evaluate our proposed framework. Also note that we obfuscate personal information in all of the datasets.

Table 2-2. Mapping of the random variables for the datasets.

Random Variable		Foursquare	Twitter	Plurk	DBLP
Candidate	y	Like	Retweet	Response	Citation
Attribute	u	User	User	User	User
	r	Tip	Tweet	Message	Paper
	c	Venue	Term	Topic	Keyword
Count	t	Likes per tip	Retweets per tweet	Responses per message	Citations per paper

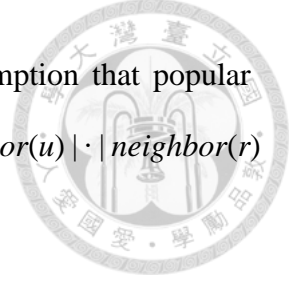


It should be noted that the unseen-type links used as ground truth are actually sparse comparing to all nodes and relations. For example, in Twitter dataset, the unseen-to-candidate ratio, $|Unseen| / (|User| \cdot |Item|)$, is merely 0.00002. Thus, predicting unseen-type links for these datasets is a very challenging task.

2.4.2 Comparing Methods

We use nine unsupervised model for comparison. The first three methods are single attribute-to-candidate functions: UF, IO, and CP. Another six methods are as follows (note that all methods are executed on the whole heterogeneous social network):

- **Betweenness Centrality (BC)**. This method is used to measure an edge's importance in a network. The BC value of an edge equals to the number of shortest paths from all nodes to all others that pass through that edge. For each candidate pair, we add a *pseudo* unseen-type link in network. Then, we generate BC values of pseudo links as their prediction scores.
- **Jaccard Coefficient (JC)**. This method is used to directly compute the relatedness of a user u to an item r , which is defined as $|neighbor(u) \cap neighbor(r)| / |neighbor(u) \cup neighbor(r)|$. This score is used to predict whether u likes r .



- **Preferential Attachment (PA).** This method bases on an assumption that popular users tends to like popular items. Therefore, it is defined as $|\text{neighbor}(u)| \cdot |\text{neighbor}(r)|$, which is used as the prediction scores.

- **Attractiveness (AT).** This method is designed to compute user-to-user attractiveness using aggregated count [61]. We transform it to predict unseen-type links. It first computes owner-item attractiveness P_{vr} from owner v to item r as

$$P_{vr} = \frac{\sigma(r, \Phi)}{\sum_{c(r')=c(r)} \sigma(r', \Phi)} \quad (11)$$

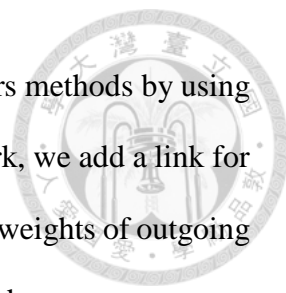
where Φ is the set of “like” links, and $\sigma(r, \Phi)$ is the aggregative statistic of item r , as defined in Section 2.2. Then, it compute the user-owner attractiveness P_{uv} from user u to v as

$$P_{uv} = 1 - \prod_r (g_{uv} \cdot (1 - P_{vr})) \quad (12)$$

where $g_{uv} = 1$ if u and v are friends, otherwise 0. To perform link prediction, we further compute user-item attractiveness P_{ur} (the probability of user u likes item r) as

$$P_{ur} = P_{uv} \cdot P_{vr} \quad (13)$$

- **PageRank with Priors (PRP).** This method executes PageRank algorithm [59] for $|R|$ times, once for each item. For specific item r , we set the prior of the item node to 1, and priors of all other nodes to 0. Thus, the probability of user u likes item r is modeled using PageRank score of the user node u . We set the random restart probability as 0.15.

- 
- **AT-PRP.** We combine the Attractiveness and PageRank with Priors methods by using the weight of the links. That is, in the heterogeneous social network, we add a link for each $\langle u, r \rangle$ pair, with weight equals to P_{ur} . We then normalize all weights of outgoing links to sum up to 1, and run PageRank with Priors as mentioned above.

2.4.3 Settings

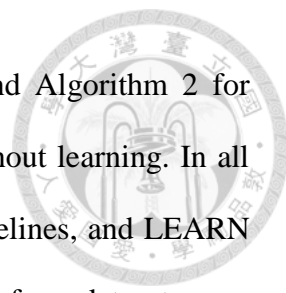
Because of the sparsity of unseen-links in ground-truth, we use Area Under ROC Curve (AUC) [9] [36] and Normalized Discounted Cumulative Gain (NDCG) [23] to evaluate our proposed method. For each item, we rank all the candidate pairs based on their predicted positive marginal probabilities, and then compare the rankings with the ground-truths to obtain AUC and NDCG scores. Finally, we average the scores over all items.

We select Loopy Belief Propagation (LBP) as our base inference method [46], utilize MALLET [42] for LBP inference, and apply LingPipe [2] for stemming. We use JUNG [45] to compute betweenness centrality and PageRank with Priors algorithms.

In FGM-AS, we set all zero potential function values to a small constant (0.000001), and use learning rate $\eta = 0.0001$. We run all experiments on a Linux server with AMD Opteron 2350 2.0GHz Quad-core CPU and 32GB memory.

2.4.4 Results

The results of different methods using AUC and NDCG are shown in Table 2-3. The



LEARN method is to exploit Algorithm 1 to perform learning and Algorithm 2 for inference, while INFER is to exploit Algorithm 2 for inference without learning. In all cases, LEARN performs best. Note that INFER outperforms all baselines, and LEARN provides further improvement than INFER. Averaging over the four datasets, our framework (LEARN) are 9.79% AUC and 12.81% NDCG better than the best comparing methods. LEARN achieves best result for Foursquare dataset, with improvement of 16.15% in AUC and 29.60% in NDCG.

From Table 2-3, we see that the performance distinction between the three attribute-to-candidate functions, UF, IO, and CP, varies depending on the dataset used. We believe that these three functions are complementary to each other, and can be ensembled to contribute to our integrated framework. BC does not work well in all experiments, JC performs well for Twitter in terms of NDCG, and PA performs well for DBLP in terms of AUC. On the other hand, AT is in general the strongest comparing method (performs best among comparing methods in both metrics for all four datasets); PRP in general does not perform well; AT-PRP ranks just between AT and PRP. Our framework consistently outperforms these comparing methods significantly. Based on the above experiment results, we believe our framework can be a general method to solve the unseen-type link prediction problem.



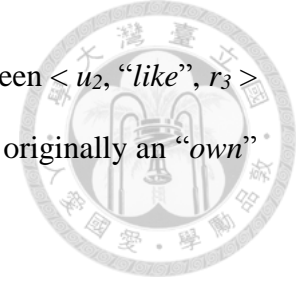
Table 2-3. Experiment results of our framework (FGM-AS) and all comparing methods (in percentage).

Method	Foursquare		Twitter		Plurk		DBLP	
	AUC	NDCG	AUC	NDCG	AUC	NDCG	AUC	NDCG
UF	76.74	21.66	73.49	18.87	71.08	35.01	70.28	25.07
IO	81.31	51.60	69.98	18.93	69.86	35.33	68.51	23.84
CP	74.03	20.56	67.38	17.15	70.69	36.13	69.52	24.22
BC	67.01	21.26	67.65	18.97	69.81	31.47	64.17	21.10
JC	64.30	26.75	65.65	21.05	70.05	35.40	69.96	28.24
PA	72.28	27.09	62.30	16.39	67.42	32.68	71.41	26.12
AT	82.57	44.54	76.95	20.28	69.62	39.29	70.95	28.48
PRP	57.27	17.93	62.41	16.56	69.12	33.64	61.83	21.25
AT-PRP	71.06	22.38	68.17	18.11	70.99	36.03	67.86	24.27
INFER	86.87	71.27	78.58	25.24	74.53	39.85	86.51	41.84
LEARN	98.72	81.20	80.75	26.33	74.72	42.20	86.96	41.93
Improve	16.15	29.60	3.80	5.28	3.64	2.91	15.55	13.45

2.4.5 Candidate-to-Candidate Verification

In the previous subsection, we evaluate the attribute-to-candidate functions and compare them to our proposed framework. However, the candidate-to-candidate functions cannot be evaluated independently (i.e., without attribute-to-candidate functions). Therefore, we verify the feasibility of the four functions, namely OI, FI, OF, CC, and CI, by performing a simple analysis in our datasets. First, we set all “own” links as “like” links. As shown in Figure 2-1, we set $\langle u_1, \text{“like”}, r_1 \rangle$, $\langle u_1, \text{“like”}, r_2 \rangle$, and $\langle u_2, \text{“like”}, r_3 \rangle$, as positive prediction. Then, we apply the above four candidate-to-candidate functions to extend the predicted links.

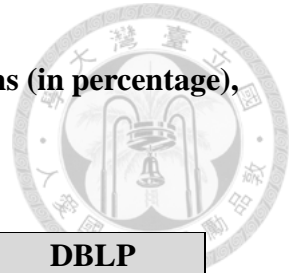
For example, considering OF function, there will be a link between $\langle u_2, \text{"like"}, r_3 \rangle$ and $\langle u_1, \text{"like"}, r_3 \rangle$. Because $\langle u_2, \text{"like"}, r_3 \rangle$ is positive (i.e., it is originally an “own” link), we predict $\langle u_1, \text{"like"}, r_3 \rangle$ as positive based on OF.



We compare the result of candidate-to-candidate functions using precision and recall with the unseen-type links in ground-truth, as shown in Table 2-4. We also ensemble the four functions and examine the effectiveness of the combination (the *All* row). All of the candidate-to-candidate functions has low precision (less than 4%), but have some extend of recall (especially *All*). For Foursquare and DBLP datasets, the recall of *All* reaches as high as 95.00% and 95.15%, respectively. It should be noted that OI performs bad for Twitter, Plurk and DBLP datasets, but provides some improvement for Foursquare dataset. On the other hand, FI seems to be of little use for Twitter dataset, but it does provide information for other three datasets. Therefore, we regard these four candidate-to-candidate functions as complementary to each other, and can be ensembled to contribute to our framework.

Table 2-4. Verification results of candidate-to-candidate functions (in percentage),

Pre. = precision, Rec. = recall.



Function	Foursquare		Twitter		Plurk		DBLP	
	Pre.	Rec.	Pre.	Rec.	Pre.	Rec.	Pre.	Rec.
OI	2.14	37.50	0.00	0.00	0.00	0.00	0.00	0.00
FI	0.33	55.00	0.00	0.00	3.25	33.55	1.53	60.68
OF	0.35	40.00	0.21	20.00	3.23	37.31	1.53	60.68
CC	0.20	2.50	0.74	20.00	1.36	18.76	2.64	86.65
CI	0.08	22.50	0.00	0.00	0.00	0.00	0.12	2.43
All	0.22	95.00	0.05	40.00	1.58	51.43	1.24	95.15

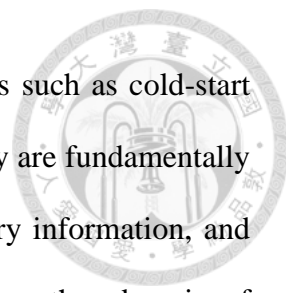


2.5 Related Work

In this subsection, we discuss some of works related to unsupervised unseen-link prediction framework using aggregative statistics.

2.5.1 Link Prediction

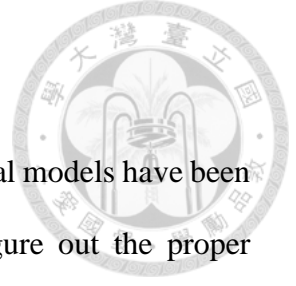
Our problem is effectively link prediction in heterogeneous social network. Link prediction is a well-studied task in social network analysis, and is characterized by graph topology, testing how *proximal* nodes are to each other [35]. Many features have been tested and developed for homogeneous network, using different graph topological properties [39]. However, such approaches do not consider the sparsity and diversity of heterogeneous social network. Feature design for heterogeneous social network was recently explored [62], casting as a supervised learning task [29]. One area of research interest is to predict actual popularity of a microblog (e.g., tweet) in a social media. In this case, the task is formulated as a supervised learning problem, where it can be binary (e.g., whether a tweet will be retweeted or not) or multi-class (e.g., assign the prediction of how a tweet will be retweeted by popularity category) classification problem [20] [47]. Another approach applies probabilistic model on social media response prediction [64]. This work essentially incorporates collaborative filtering accounting user and item (i.e., tweets) features, but still require training data. Another related area is to predict the link from user to venue (i.e., point of interest recommendation) using geographic information [63]. However, such method fails to utilize effects of information propagation in social network.



Regarding unsupervised link prediction, there have been works such as cold-start link prediction [33], transfer learning [10], and triad census [8]. They are fundamentally different from this work. Cold-start link prediction requires category information, and works only on homogeneous network. Transfer learning assumes another domain of labeled data is available. Triad census does not consider the aggregative statistics information in the networks. Pure unsupervised heterogeneous social network link prediction explores different context of the data by examining probabilistically the topological features of the reweighed path [8] [62]. However, these works usually predict links between two entities of the same type, holding the underlying assumption that birds of a feather flock together. Our work tries to predict links between two different types (usually users and items) where such assumption is not likely to hold.

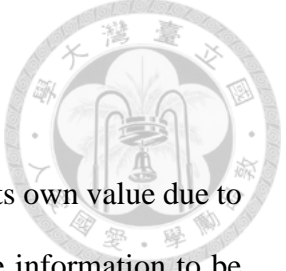
2.5.2 Factor Graph and Max-Margin Learning

Factor graph [26] is a unified framework for general probabilistic graphical models. Recently, factor graphs have been widely adopted to resolve various problems [21] [51] [55] [57]. Among these applications, factor graphs are suitable for social relationship prediction tasks. [55] proposed a time-constrained unsupervised probabilistic factor graph (TPFG) to model the advisor-advisee relationship using time information. Triad Factor Graph (TriFG) model [21] incorporates the factor graph representations and social theories over triads into a semi-supervised model. [51] investigates the relationship prediction problem on heterogeneous social networks. Previous attempts are extended and integrated into a transfer-based factor graph (TranFG) model. However, these methods either need additional external information or do not consider the aggregation of statistics during computation.



Several margin-based learning methods on probabilistic graphical models have been proposed. Previous methods require the ground-truth labels to figure out the proper direction of parameter update. For example, [53] formulates the parameter fitting problem as a quadratic program and performs Sequential Minimal Optimization (SMO) learning to solve the problem. For max-margin methods solving similar problems such as structural support vector machines [54], the ground-truth is also needed to fit these models. However, in our problem, it is the aggregative statistics instead of the ground-truth labels that are given. Therefore, our framework maximizes the *ranked-margin* instead of traditional margin.

2.6 Short Summary

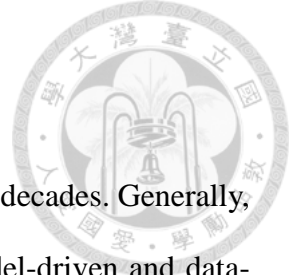


Mining on social networks using incomplete information has gained its own value due to its applicability, as in the real world we cannot always expect all the information to be observable. In this study, we demonstrate that the unseen-type link prediction can be solved using an unsupervised framework through exploiting the aggregative statistics. We showed how various information sources in the heterogeneous social network can be modeled all together in a factor graph, propose a novel learning algorithm to learn the parameters using aggregated counts, and devise an inference algorithm to predict unseen-type links using learnt parameters. With such framework, one can now derive hypotheses on the individual behavior using the group statistics. Especially, under the growing concern of personal privacy preservation, we believe our framework provides a means for applications that tries to distill personal preference information from the statistics. On the other hand, in the area of biomedicine, our framework can be applied to identify novel protein-disease relationships, given clinical aggregated observations. To summarize, in this study we propose an unsupervised framework to discover the links of unlabeled *relationship* in *heterogeneous* networks.

Chapter 3 Diffusion Prediction of Novel Topics



This study brings a marriage of two seemingly unrelated topics, natural language processing (NLP) and social network analysis (SNA). We propose a new task in SNA which is to predict the diffusion of a new topic, and design a learning-based framework to solve this problem. We exploit the latent semantic information among users, topics, and social connections as features for prediction. Our framework is evaluated on real data collected from public domain. The experiments show 16% AUC improvement over baseline methods.

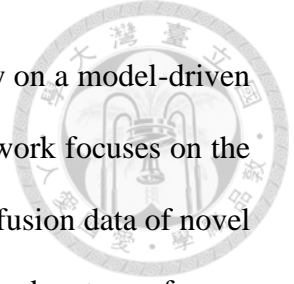


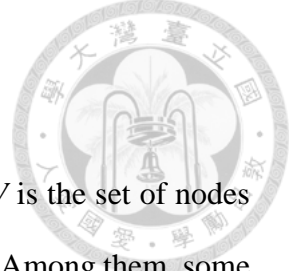
3.1 Overview

The diffusion of information on social networks has been studied for decades. Generally, the proposed strategies can be categorized into two categories, model-driven and data-driven. The model-driven strategies, such as independent cascade model [25], rely on certain manually crafted, usually intuitive, models to fit the diffusion data without using diffusion history. The data-driven strategies usually utilize learning-based approaches to predict the future propagation given historical records of prediction [13] [15] [48]. Data-driven strategies usually perform better than model-driven approaches because the past diffusion behavior is used during learning [15].

Recently, researchers started to exploit content information in data-driven diffusion models [13] [48] [67]. However, most of the data-driven approaches assume that in order to train a model and predict the future diffusion of a topic, it is required to obtain historical records about how this topic has propagated in a social network [48] [67]. We argue that such assumption does not always hold in the real-world scenario, and being able to forecast the propagation of novel or unseen topics is more valuable in practice. For example, a company would like to know which users are more likely to be the source of “viva voce” of a *newly* released product for advertising purpose. A political party might want to estimate the potential degree of responses of a half-baked policy before deciding to bring it up to public. To achieve such goal, it is required to predict the future propagation behavior of a topic even *before* any actual diffusion happens on this topic (i.e., no historical propagation data of this topic are available). Lin et al. also propose an idea aiming at predicting the inference of implicit diffusions for novel topics [38]. The main difference between their work and ours is that they focus on implicit diffusions,

whose data are usually not available. Consequently, they need to rely on a model-driven approach instead of a data-driven approach. On the other hand, our work focuses on the prediction of explicit diffusion behaviors. Despite the fact that no diffusion data of novel topics is available, we can still design a data-driven approach taking advantage of some explicit diffusion data of known topics. Our experiments show that being able to utilize such information is critical for diffusion prediction.





3.2 The Novel-Topic Diffusion Model

We start by assuming an existing social network $G = (V, E)$, where V is the set of nodes (or user) v , and E is the set of link e . The set of topics is denoted as T . Among them, some are considered as novel topics (denoted as N), while the rest (R) are used as the training records. We are also given a set of diffusion records $D = \{d \mid d = (src, dest, t)\}$, where src is the source node (or diffusion source), $dest$ is the destination node, and t is the topic of the diffusion that belongs to R but not N . We assume that diffusions cannot occur between nodes without direct social connection; any diffusion pair implies the existence of a link $e = (src, dest) \in E$. Finally, we assume there are sets of keywords or tags that relevant to *each* topic (including existing and novel topics). Note that the set of keywords for novel topics should be seen in that of existing topics. From these sets of keywords, we construct a topic-word matrix $TW = (P(word_j \mid topic_i))_{i,j}$ of which the elements stand for the conditional probabilities that a word appears in the text of a certain topic. Similarly, we also construct a user-word matrix $UW = (P(word_j \mid user_i))_{i,j}$ from these sets of keywords. Given the above information, the goal is to predict whether a given link is active (i.e., belongs to a diffusion link) for topics in N .

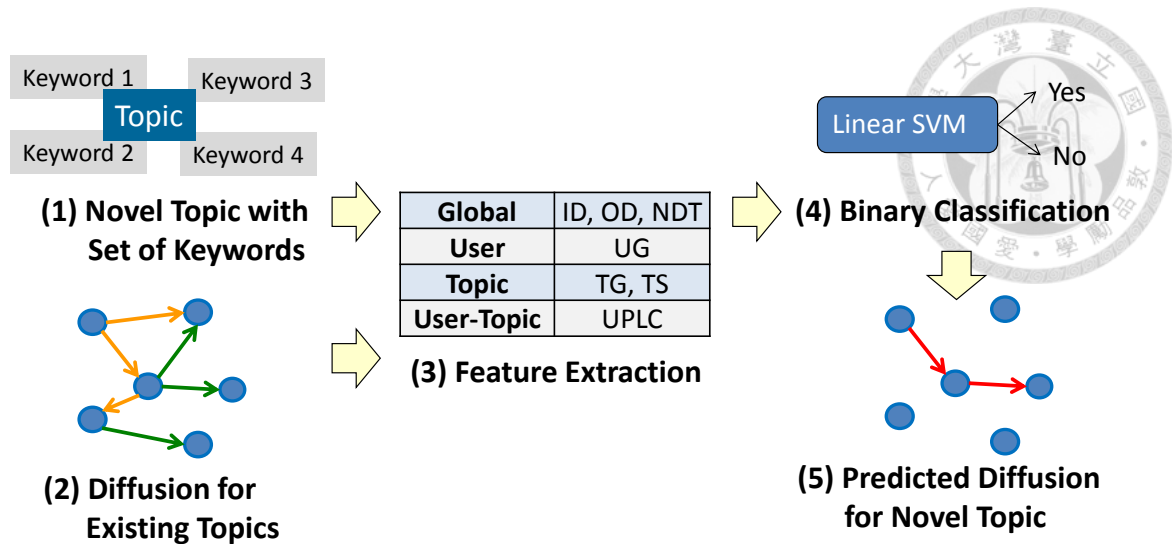


Figure 3-1. The novel-topic diffusion model.

3.2.1 The Framework

The main challenge of this problem lays in that the past diffusion behaviors of new topics are missing. To address this challenge, we propose a supervised diffusion discovery framework (Figure 3-1) that exploits the latent semantic information among users, topics, and their explicit / implicit interactions. We take (1) the novel topic with a set of keywords describing the topic, and (2) diffusion for existing topics as inputs. Next, we extract features, and finally perform binary classification to predict diffusions for novel topic. Intuitively, four kinds of information are useful for prediction:

- *Topic information:* Intuitively, knowing the signatures of a topic (e.g., is it about politics?) is critical to the success of the prediction.
- *User information:* The information of a user such as the personality (e.g., whether this user is aggressive or passive) is generally useful.



- *User-topic interaction*: Understanding the users' preference on certain topics can improve the quality of prediction.
- *Global information*: We include some global features (e.g., topology info) of social network.

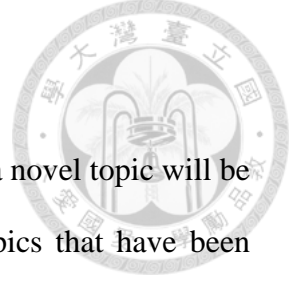
Below we will describe how these four kinds of information can be modeled in our framework.

3.2.2 Topic Information

We extract hidden topic category information to model *topic signature*. In particular, we exploit the Latent Dirichlet Allocation (LDA) method [5], which is a widely used topic modeling technique, to decompose the topic-word matrix TW into hidden topic categories:

$$TW = TH * HW \quad (14)$$

, where TH is a topic-hidden matrix, HW is hidden-word matrix, and h is the manually-chosen parameter to determine the size of hidden topic categories. TH indicates the distribution of each topic to hidden topic categories, and HW indicates the distribution of each lexical term to hidden topic categories. Note that TW and TH include both existing and novel topics. We utilize $TH_{t,*}$, the row vector of the topic-hidden matrix TH for a topic t , as a feature set. In brief, we apply LDA to extract the topic-hidden vector $TH_{t,*}$ to model *topic signature* (TG) for both existing and novel topics.



Topic information can be further exploited. To predict whether a novel topic will be propagated through a link, we can first enumerate the existing topics that have been propagated through this link. For each such topic, we can calculate its similarity with the new topic based on the hidden vectors generated above (e.g., using cosine similarity between feature vectors). Then, we sum up the similarity values as a new feature: *topic similarity (TS)*. For example, a link has previously propagated two topics for a total of three times {ACL, KDD, ACL}, and we would like to know whether a new topic, EMNLP, will propagate through this link. We can use the topic-hidden vector to generate the similarity values between EMNLP and the other topics (e.g., {0.6, 0.4, 0.6}), and then sum them up (1.6) as the value of *TS*.

3.2.3 User Information

Similar to topic information, we extract latent personal information to model *user signature* (the users are anonymized already). We apply LDA on the user-word matrix UW :

$$UW = UM * MW \quad (15)$$

, where UM is the user-hidden matrix, MW is the hidden-word matrix, and m is the manually-chosen size of hidden user categories. UM indicates the distribution of each user to the hidden user categories (e.g., age). We then use $UM_{u,*}$, the row vector of UM for the user u , as a feature set. In brief, we apply LDA to extract the user-hidden vector $UM_{u,*}$ for both source and destination nodes of a link to model *user signature (UG)*.



3.2.4 User-Topic Interaction

Modeling user-topic interaction turns out to be non-trivial. It is not useful to exploit latent semantic analysis directly on the user-topic matrix $UR = UQ * QR$, where UR represents *how many times each user is diffused for existing topic R ($R \in T$)*, because UR does not contain information of novel topics, and neither do UQ and QR . Given no propagation record about novel topics, we propose a method that allows us to still extract implicit user-topic information. First, we extract from the matrix TH (described in Section 3.2) a subset RH that contains only information about existing topics. Next we apply left division to derive another user-hidden matrix UH :

$$UH = (RH \setminus UR^T)^T = ((RH^T RH)^{-1} RH^T UR^T)^T \quad (16)$$

Using left division, we generate the UH matrix using existing topic information. Finally, we exploit $UH_{u,*}$, the row vector of the user-hidden matrix UH for the user u , as a feature set.

Note that novel topics were included in the process of learning the hidden topic categories on RH ; therefore the features learned here do implicitly utilize some latent information of novel topics, which is not the case for UM . Experiments confirm the superiority of our approach. Furthermore, our approach ensures that the hidden categories in topic-hidden and user-hidden matrices are identical. Intuitively, our method directly models the user's preference to topics' signature (e.g., how capable is this user to propagate topics in politics category?). In contrast, the UM mentioned in Section 3.3 represents the users' signature (e.g., aggressiveness) and has nothing to do with their

opinions on a topic. In short, we obtain the user-hidden probability vector $UH_{u,*}$ as a feature set, which models *user preferences to latent categories (UPLC)*.



3.2.5 Global Features

Given a candidate link, we can extract global social features such as *in-degree (ID)* and *out-degree (OD)*. We tried other features such as PageRank values but found them not useful. Moreover, we extract the *number of distinct topics (NDT)* for a link as a feature. The intuition behind this is that the more distinct topics a user has diffused to another, the more likely the diffusion will happen for novel topics.

3.2.6 Complexity Analysis

The complexity to produce each feature is as below:

- (1) *Topic information*: $O(I * |T| * h * B_t)$ for LDA using Gibbs sampling, where I is # of the iterations in sampling, $|T|$ is # of topics, and B_t is the average # of tokens in a topic.
- (2) *User information*: $O(I * |V| * m * B_u)$, where $|V|$ is # of users, and B_u is the average # of tokens for a user.
- (3) *User-topic interaction*: the time complexity is $O(h^3 + h^2 * |T| + h * |T| * |V|)$.
- (4) *Global features*: $O(|D|)$, where $|D|$ is # of diffusions.



3.3 Experiments

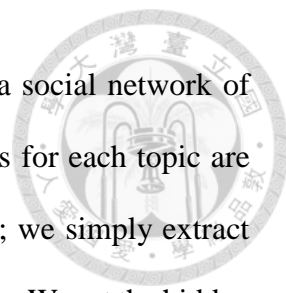
For evaluation, we try to use the diffusion records of old topics to predict whether a diffusion link exists between two nodes given a new topic.

3.3.1 Dataset and Evaluation Metric

We first identify 100 most popular topics (e.g., earthquake) in Plurk from 01/2011 to 05/2011. Plurk is a popular micro-blog service in Asia with more than 5 million users [27]. We manually separate the 100 topics into 7 groups. We use topic-wise 4-fold cross validation to evaluate our method, because there are only 100 available topics. For each group, we select 3/4 of the topics as training and 1/4 as validation. For validation set we remove diffusions not mentioned in training set.

The positive diffusion records are generated based on the post-response behavior. That is, if a person x posts a message containing one of the selected topic t , and later there is a person y responding to this message, we consider a diffusion of t has occurred from x to y (i.e., (x, y, t) is a positive instance). Our dataset contains a total of 1,146,995 positive instances out of 100 distinct topics; the largest and smallest topic contains 210,745 and 1,644 diffusions, respectively. Also, the same amount of negative instances for each topic (totally 1,146,995) is sampled for binary classification (similar to the setup in KDD Cup 2011 Track 2). The negative links of a topic t are sampled randomly based on the absence of responses for that given topic.

The underlying social network is created using the post-response behavior as well. We assume there is an acquaintance link between x and y if and only if x has responded



to y (or vice versa) on at least one topic. Eventually we generated a social network of 163,034 nodes and 382,878 links. Furthermore, the sets of keywords for each topic are required to create the TW and UW matrices for latent topic analysis; we simply extract the content of posts and responses for each topic to create both matrices. We set the hidden category number $h = m = 7$, which is equal to the number of topic groups.

We use area under ROC curve (AUC) to evaluate our proposed framework [9]; we rank the testing instances based on their likelihood of being positive, and compare it with the ground truth to compute AUC.

3.3.2 Implementation and Baseline

After trying many classifiers and obtaining similar results for all of them, we report only results from LIBLINEAR with $c=0.0001$ [12] due to space limitation. We remove stop-words, use SCWS [19] for tokenization, and MALLET [42] and GibbsLDA++ [49] for LDA.

There are three baseline models we compare the result with. First, we simply use the total number of existing diffusions among all topics between two nodes as the single feature for prediction. Second, we exploit the independent cascading model [25], and utilize the normalized total number of diffusions as the propagation probability of each link. Third, we try the heat diffusion model [41], set initial heat proportional to out-degree, and tune the diffusion time parameter until the best results are obtained. Note that we did not compare with any data-driven approaches, as we have not identified one that can predict diffusion of novel topics.



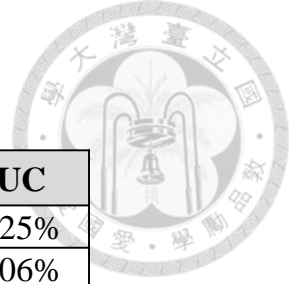
3.3.3 Results

The result of each model is shown in Table 3-1. All except two features outperform the baseline. The best single feature is *TS*. Note that *UPLC* performs better than *UG*, which verifies our hypothesis that maintaining the same hidden features across different LDA models is better. We further conduct experiments to evaluate different combinations of features (Table 3-2), and found that the best one (*TS + ID + NDT*) results in about 16% improvement over the baseline, and outperforms the combination of all features. As stated in [60], adding useless features may cause the performance of classifiers to deteriorate. Intuitively, *TS* captures both latent topic and historical diffusion information, while *ID* and *NDT* provide complementary social characteristics of users.

Table 3-1. Single-feature results.

Method	Feature	AUC
Baseline	Existing Diffusion	58.25%
	Independent Cascade	51.53%
	Heat Diffusion	56.08%
Learning	Topic Signature (<i>TG</i>)	50.80%
	Topic Similarity (<i>TS</i>)	69.93%
	User Signature (<i>UG</i>)	56.59%
	User Preferences to Latent Categories (<i>UPLC</i>)	61.33%
	In-degree (<i>ID</i>)	65.55%
	Out-degree (<i>OD</i>)	59.73%
	Number of Distinct Topics (<i>NDT</i>)	55.42%

Table 3-2. Feature combination results.



Method	Feature	AUC
Baseline	Existing Diffusion	58.25%
Learning	ALL	65.06%
	<i>TS + UPLC + ID + NDT</i>	67.67%
	<i>TS + UPLC + ID</i>	64.80%
	<i>TS + UPLC + NDT</i>	66.01%
	<i>TS + ID + NDT</i>	73.95%
	<i>UPLC + ID + NDT</i>	67.24%

3.4 Short Summary



The main contributions of this study are as below:

- (1) We propose a novel task of predicting the diffusion of unseen topics, which has wide applications in real-world.
- (2) Compared to the traditional model-driven or content-independent data-driven works on diffusion analysis, our solution demonstrates how one can bring together ideas from two different but promising areas, NLP and SNA, to solve a challenging problem.
- (3) Promising experiment result (74% in AUC) not only demonstrates the usefulness of the proposed models, but also indicates that predicting diffusion of unseen topics without historical diffusion data is feasible.

To summarize, in this study we propose a supervised learning framework to discover the links of unlabeled *diffusion* in *homogeneous* networks.

Chapter 4 Conclusion



In this dissertation, we investigate two dimensions of the link discovery with unlabeled data problem: (1) link prediction using aggregative statistics, and (2) diffusion prediction of novel topics. For each problem, we devise a learning-based frameworks to integrate the diverse information and solve discover the links. Furthermore, we conduct experiments on real-world datasets (Foursquare, Twitter, Plurk, DBLP), and the results show that our proposed frameworks provide reasonably high performance and can solve the unlabeled link prediction problems.

A plausible future direction is to consider the opinion (e.g., “positive” or “negative”) of the links to be predicted. An example is to predict the “dislike” link instead of “like”; the intuitions behind “dislike” may not simply be the inverse of “like”. Another example is that although two topics are highly related under the computation of LDA, they might be opposite or competitive to each other (e.g., different mobile phone companies or different politic parties); thus the diffusion prediction process may also be influenced by the opinion. In this dissertation we mainly consider the “positive” links, therefore including the idea of opinion mining may further improve the prediction results.

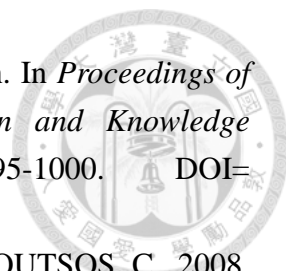
Another consideration is the efficiency of the proposed algorithms. In the big-data era, the data are increasing rapidly, and may require shorter computation time to ensure the effectiveness of the prediction results. However, our proposed methods (e.g., FGM-AS or LDA-based classification) are computation-intensive, especially for large-scale

datasets or rapid online data. Therefore, a natural extension of this dissertation is to fasten the computation process. One plausible method is divide-and-conquer scheme. That is, cluster the data in to smaller but to some extent independent groups (e.g., divide the Foursquare data into smaller geographical districts), and then compute each group in parallel using the state-of-the-art distributed or GPU-based computing approaches. We believe such methods can alleviate the issue of computational overhead.


Bibliography





- [1] ADAMIC, L.A. and ADAR, E., 2003. Friends and Neighbors on the Web. *Social Networks* 25, 3, 211--230.
- [2] ALIAS-I, 2008. LingPipe 4.1.0.
- [3] BARABASI, A.L. and ALBERT, R., 1999. Emergence of Scaling in Random Networks. *Science* 286, 5439, 509.
- [4] BILGIC, M., NAMATA, G.M., and GETOOR, L., 2007. Combining Collective Classification and Link Prediction. In *Proceedings of the 7th IEEE International Conference on Data Mining Workshops (ICDMW)* (2007), 1336107, 381-386. DOI= <http://dx.doi.org/10.1109/icdmw.2007.28>.
- [5] BLEI, D.M., NG, A.Y., and JORDAN, M.I., 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research (JMLR)* 3, 993-1022.
- [6] BRIN, S. and PAGE, L., 1998. The Anatomy of a Large-Scale Hypertextual Web Search Engine. *Computer Networks and ISDN Systems* 30, 1--7, 107-117.
- [7] CHEN, M.-Y., LIN, H.-N., SHIH, C.-A., HSU, Y.-C., HSU, P.-Y., and HSIEH, S.-K., 2010. Classifying Mood in Plurks. In *Proceedings of the 22nd Conference on Computational Linguistics and Speech Processing (ROCLING)*.
- [8] DAVIS, D., LICHTENWALTER, R., and CHAWLA, N.V., 2011. Multi-relational Link Prediction in Heterogeneous Information Networks. In *Proceedings of the 2011 International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 281-288. DOI= <http://dx.doi.org/10.1109/asonam.2011.107>.
- [9] DAVIS, J. and GOADRICH, M., 2006. The Relationship Between Precision-Recall and ROC Curves. In *Proceedings of the 23rd International Conference on Machine Learning (ICML)* (2006), 1143874, 233-240. DOI= <http://dx.doi.org/10.1145/1143844.1143874>.
- [10] DONG, Y., TANG, J., WU, S., TIAN, J., CHAWLA, N.V., RAO, J., and CAO, H., 2012. Link Prediction and Recommendation across Heterogeneous Social Networks. In *Proceedings of the 12th IEEE International Conference on Data Mining (ICDM)* (2012), 181 -190.
- [11] DOPPA, J.R., YU, J., TADEPALLI, P., and GETOOR, L., 2009. Chance-Constrained Programs for Link Prediction. In *Proceedings of the NIPS Workshop on Analyzing Networks and Learning with Graphs* (2009).
- [12] FAN, R.-E., CHANG, K.-W., HSIEH, C.-J., WANG, X.-R., and LIN, C.-J., 2008. LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research (JMLR)* 9, 1871-1874.
- [13] FEI, H., JIANG, R., YANG, Y., LUO, B., and HUAN, J., 2011. Content Based

- 
- Social Behavior Prediction: a Multi-Task Learning Approach. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM)* (2011), 2063719, 995-1000. DOI=<http://dx.doi.org/10.1145/2063576.2063719>.
- [14] GALLAGHER, B., TONG, H., ELIASSI-RAD, T., and FALOUTSOS, C., 2008. Using Ghost Edges for Classification in Sparsely Labeled Networks. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)* (2008), 1401925, 256-264. DOI=<http://dx.doi.org/10.1145/1401890.1401925>.
- [15] GALUBA, W., ABERER, K., CHAKRABORTY, D., DESPOTOVIC, Z., and KELLERER, W., 2010. Outtweeting the Twitterers - Predicting Information Cascades in Microblogs. In *Proceedings of the 3rd Conference on Online Social Networks* (2010), 1863193, 3-3.
- [16] GETOOR, L. and DIEHL, C.P., 2005. Link Mining: A Survey. *ACM SIGKDD Explorations Newsletter* 7, 2, 3-12. DOI=<http://dx.doi.org/10.1145/1117454.1117456>.
- [17] GUEORGI, K., 2006. *Effects of Missing Data in Social Networks*. Elsevier, Amsterdam, PAYS-BAS.
- [18] HASAN, M.A., CHAOJI, V., SALEM, S., and ZAKI, M., 2006. Link Prediction Using Supervised Learning. In *Proceedings of the SDM Workshop on Link Analysis, Counterterrorism and Security* (2006), hasan2006prediction.
- [19] HIGHTMAN, 2012. Simple Chinese Words Segmentation (SCWS).
- [20] HONG, L., DAN, O., and DAVISON, B.D., 2011. Predicting popular messages in Twitter. In *Proceedings of the 20th International Conference on World Wide Web (WWW)* (2011), 1963222, 57-58. DOI=<http://dx.doi.org/10.1145/1963192.1963222>.
- [21] HOPCROFT, J., LOU, T., and TANG, J., 2011. Who Will Follow You Back? Reciprocal Relationship Prediction. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM)* (2011), 1137-1146.
- [22] JEH, G. and WIDOM, J., 2002. SimRank: A Measure of Structural-Context Similarity. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)* (2002), 775126, 538-543. DOI=<http://dx.doi.org/10.1145/775047.775126>.
- [23] JRVELIN, K. and KEKLINEN, J., 2002. Cumulated Gain-Based Evaluation of IR Techniques. *ACM Transactions on Information Systems* 20, 4, 422-446. DOI=<http://dx.doi.org/10.1145/582415.582418>.
- [24] KATZ, L., 1953. A New Status Index Derived from Sociometric Analysis.

- Psychometrika* 18, 1, 39-43.
- [25] KEMPE, D., KLEINBERG, J., and TARDOS, E., 2003. Maximizing the Spread of Influence Through a Social Network. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)* (2003), 956769, 137-146. DOI= <http://dx.doi.org/10.1145/956750.956769>.
- [26] KSCHISCHANG, F.R., FREY, B.J., and LOELIGER, H.-A., 2001. Factor Graphs and the Sum-Product Algorithm. *IEEE TRANSACTIONS ON INFORMATION THEORY* 47, 2.
- [27] KUO, T.-T., HUNG, S.-C., LIN, W.-S., LIN, S.-D., PENG, T.-C., and SHIH, C.-C., 2011. Assessing the Quality of Diffusion Models Using Real-World Social Network Data. In *Proceedings of the 2011 International Conference on Technologies and Applications of Artificial Intelligence (TAAI)* (2011).
- [28] KUO, T.-T., HUNG, S.-C., LIN, W.-S., PENG, N., LIN, S.-D., and LIN, W.-F., 2012. Exploiting Latent Information to Predict Diffusions of Novel Topics on Social Networks. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers - Volume 2* (2012), 2390743, 344-348.
- [29] KUO, T.-T. and LIN, S.-D., 2011. Learning-Based Concept-Hierarchy Refinement Through Exploiting Topology, Content and Social Information. *Information Sciences* 181, 12, 2512-2528. DOI= <http://dx.doi.org/DOI:10.1016/j.ins.2011.02.006>.
- [30] KUO, T.-T., YAN, R., HUANG, Y.-Y., KUNG, P.-H., and LIN, S.-D., 2013. Unsupervised Link Prediction Using Aggregative Statistics on Heterogeneous Social Networks. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)* (2013).
- [31] KUO, T.-T., YEH, J.-J., LIN, C.-J., and LIN, S.-D., 2010. Designing, Analyzing and Exploiting Stake-Based Social Networks. In *Proceedings of the 2010 International Conference on Advances in Social Networks Analysis and Mining (ASONAM)* (2010), 1900783, 402-403. DOI= <http://dx.doi.org/10.1109/asonam.2010.14>.
- [32] KUO, T.-T., YEH, J.-J., LIN, C.-J., and LIN, S.-D., 2010. StakeNet: Devise, Study and Utilize Social Networks Using Stakeholder Information. In *Proceedings of the 2010 International Conference on Technologies and Applications of Artificial Intelligence (TAAI)* (2010), 1935537, 86-93. DOI= <http://dx.doi.org/10.1109/taai.2010.25>.
- [33] LEROY, V., CAMBAZOGLU, B.B., and BONCHI, F., 2010. Cold Start Link Prediction. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)* (2010), 393-402.

- 
- [34] LEY, M., 2002. The DBLP Computer Science Bibliography: Evolution, Research Issues, Perspectives. In *SPIRE*.
- [35] LIBEN-NOWELL, D. and KLEINBERG, J., 2007. The Link-Prediction Problem for Social Networks. *Journal of the American society for information science and technology* 58, 7, 1019-1031.
- [36] LICHTENWALTER, R. and CHAWLA, N.V., 2012. Link Prediction: Fair and Effective Evaluation. In *ASONAM*, 376-383.
- [37] LICHTENWALTER, R.N., LUSSIER, J.T., and CHAWLA, N.V., 2010. New Perspectives and Methods in Link Prediction. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)* (2010), 1835837, 243-252. DOI=<http://dx.doi.org/10.1145/1835804.1835837>.
- [38] LIN, C.X., MEI, Q.Z., JIANG, Y.L., HAN, J.W., and QI, S.X., 2011. Inferring the Diffusion and Evolution of Topics in Social Communities. In *Proceedings of the IEEE International Conference on Data Mining*.
- [39] LU, L. and ZHOU, T., 2011. Link Prediction in Complex Networks: A Survey *Physica A: Statistical Mechanics and its Applications* 390, 6, 1150-1170.
- [40] LU, Z., SAVAS, B., TANG, W., and DHILLON, I.S., 2010. Supervised Link Prediction Using Multiple Sources. In *Proceedings of the 10th IEEE International Conference on Data Mining (ICDM)* (2010), 1934599, 923-928. DOI=<http://dx.doi.org/10.1109/icdm.2010.112>.
- [41] MA, H., YANG, H., LYU, M.R., and KING, I., 2008. Mining Social Networks Using Heat Diffusion Processes for Marketing Candidates Selection. In *Proceedings of the 17th ACM International Conference on Information and Knowledge Management (CIKM)* (2008), 1458115, 233-242. DOI=<http://dx.doi.org/10.1145/1458082.1458115>.
- [42] MCCALLUM, A.K., 2002. MALLET: A Machine Learning for Language Toolkit.
- [43] NEWMAN, M.E., 2001. Clustering and Preferential Attachment in Growing Networks. *Physical Review E* 64, 2, 025102.
- [44] NEWMAN, M.E.J. and GIRVAN, M., 2004. Finding and Evaluating Community Structure in Networks. *Physical Review E* 69, 026113.
- [45] O'MADADHAIN, J., FISHER, D., WHITE, S., and BOEY, Y., 2003. *The JUNG (Java Universal Network/Graph) Framework*.
- [46] PEARL, J., 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.
- [47] PETROVIC, S., OSBORNE, M., and LAVRENKO, V., 2011. Rt to Win! Predicting Message Propagation in Twitter. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM)* (2011).

- 
- [48] PETROVIC, S., OSBORNE, M., and LAVRENKO, V., 2011. RT to Win! Predicting Message Propagation in Twitter. In *International AAAI Conference on Weblogs and Social Media*.
- [49] PHAN, X.-H. and NGUYEN, C.-T., 2007. GibbsLDA++: A C/C++ implementation of latent Dirichlet allocation (LDA).
- [50] ROGER, G. and MARTA, S.-P., 2009. *Missing and Spurious Interactions and the Reconstruction of Complex Networks*. National Academy of Sciences, Washington, DC, ETATS-UNIS.
- [51] TANG, J., LOU, T., and KLEINBERG, J., 2012. Inferring Social Ties Across Heterogenous Networks. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining (WSDM) (2012)*, 743-752.
- [52] TANG, J., ZHANG, J., YAO, L., LI, J., ZHANG, L., and SU, Z., 2008. ArnetMiner: Extraction and Mining of Academic Social Networks. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD) (2008)*, 1402008, 990-998. DOI= <http://dx.doi.org/10.1145/1401890.1402008>.
- [53] TASKAR, B., GUESTIN, C., and KOLLER, D., 2004. Max-Margin Markov Networks. In *Proceedings of the 2003 Conference on Advances in Neural Information Processing Systems (NIPS) (2004)*, 25.
- [54] TSOCHANTARIDIS, I., JOACHIMS, T., HOFMANN, T., and ALTUN, Y., 2005. Large Margin Methods for Structured and Interdependent Output Variables. *Journal of Machine Learning Research (JMLR) 6*, 1453-1484.
- [55] WANG, C., HAN, J., JIA, Y., TANG, J., ZHANG, D., YU, Y., and GUO, J., 2010. Mining Advisor-Advisee Relationships from Research Publication Networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD) (2010)*, 203-212.
- [56] WANG, C., SATULURI, V., and PARTHASARATHY, S., 2007. Local Probabilistic Models for Link Prediction. In *Proceedings of the 7th IEEE International Conference on Data Mining (ICDM) (2007)*, 1442084, 322-331. DOI= <http://dx.doi.org/10.1109/icdm.2007.108>.
- [57] WANG, Z., LI, J., WANG, Z., and TANG, J., 2012. Cross-Lingual Knowledge Linking Across Wiki Knowledge Bases. In *Proceedings of the 21st International Conference on World Wide Web (WWW) (2012)*, ACM, 459-468.
- [58] WATTS, D.J. and STROGATZ, S.H., 1998. Collective Dynamics of Small-World Networks. *Nature 393*, 6684, 440-442. DOI= <http://dx.doi.org/10.1038/30918>.
- [59] WHITE, S. and SMYTH, P., 2003. Algorithms for Estimating Relative Importance in Networks. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD) (2003)*, 266-275.

- 
- [60] WITTEN, I.H., FRANK, E., and HALL, M.A., 2011. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco.
- [61] WU, H.-H. and YEY, M.-Y., 2013. Influential Nodes in One-Wave Diffusion Model for Location-Based Social Networks. In *Proc. of the 17th Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD-2013)*.
- [62] YANG, Y., CHAWLA, N.V., SUN, Y., and HAN, J., 2012. Link Prediction in Heterogeneous Networks: Influence and Time Matters. In *Proceedings of the 12th IEEE International Conference on Data Mining (ICDM) (2012)*.
- [63] YE, M., YIN, P., LEE, W.-C., and LEE, D.-L., 2011. Exploiting Geographical Influence for Collaborative Point-of-Interest Recommendation. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR) (2011)*, 2009962, 325-334. DOI=<http://dx.doi.org/10.1145/2009916.2009962>.
- [64] ZAMAN, T.R., HERBRICH, R., VAN GAEL, J., and STERN, D., 2010. Predicting information spreading in twitter. In *Proceedings of the NIPS Workshop on Computational Social Science and the Wisdom of Crowds (2010)*, 17599-17601.
- [65] ZHANG, Q.-M., SHANG, M.-S., and LU, L., 2010. Similarity-Based Classification in Partially Labeled Networks. *International Journal of Modern Physics*.
- [66] ZHOU, T., LU, L., and ZHANG, Y.-C., 2009. Predicting Missing Links via Local Information. *The European Physical Journal B - Condensed Matter and Complex Systems* 71, 4, 623-630.
- [67] ZHU, J., XIONG, F., PIAO, D., LIU, Y., and ZHANG, Y., 2011. Statistically Modeling the Effectiveness of Disaster Information in Social Media. In *Proceedings of the 2011 IEEE Global Humanitarian Technology Conference*, 431-436. DOI=<http://dx.doi.org/10.1109/ghhc.2011.48>.