

Organizational Chart Inference

Jiawei Zhang*
University of Illinois at Chicago
Chicago, IL, USA
jzhan9@uic.edu

Philip S. Yu
University of Illinois at
Chicago, Chicago, IL, USA
Institute for Data Science,
Tsinghua University, China
psyu@cs.uic.edu

Yuanhua Lv
Microsoft Research
Redmond, WA, USA
yuanhual@microsoft.com

ABSTRACT

Nowadays, to facilitate the communication and cooperation among employees, a new family of online social networks has been adopted in many companies, which are called the “enterprise social networks” (ESNs). ESNs can provide employees with various professional services to help them deal with daily work issues. Meanwhile, employees in companies are usually organized into different hierarchies according to the relative ranks of their positions. The company internal management structure can be outlined with the organizational chart visually, which is normally confidential to the public out of the privacy and security concerns. In this paper, we want to study the IOC (Inference of Organizational Chart) problem to identify company internal organizational chart based on the heterogeneous online ESN launched in it. IOC is very challenging to address as, to guarantee smooth operations, the internal organizational charts of companies need to meet certain structural requirements (about its depth and width). To solve the IOC problem, a novel unsupervised method CREATE (ChArT REcovEr) is proposed in this paper, which consists of 3 steps: (1) social stratification of ESN users into different social classes, (2) supervision link inference from managers to subordinates, and (3) consecutive social classes matching to prune the redundant supervision links. Extensive experiments conducted on real-world online ESN dataset demonstrate that CREATE can perform very well in addressing the IOC problem.

Categories and Subject Descriptors

H.2.8 [Database Management]: Database Applications-Data Mining

Keywords

Organizational Chart Inference; Enterprise Social Network; Data Mining

*This work was partially done when the first author was on a summer internship at Microsoft Research.

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

KDD '15, August 11 - 14, 2015, Sydney, NSW, Australia
©2015 ACM. ISBN 978-1-4503-3664-2/15/08 ... \$15.00.
DOI: <http://dx.doi.org/10.1145/2783258.2783266>.

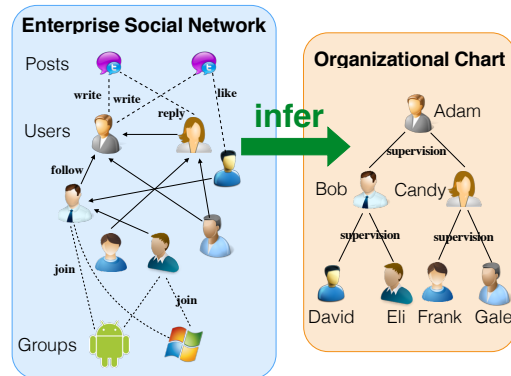


Figure 1: An example of organizational chart inference from online ESN.

1. INTRODUCTION

In social sciences, people in social organizations (e.g., a country or a company) can be categorized into different rankings of socioeconomic tiers based on factors like wealth, income, social status, occupation, power, etc. In this paper, we will take “company” as an example and the internal hierarchical structure of employees in a company can be outlined with *company organizational chart* formally. Most company organizational charts are usually tree-structure diagrams with CEO at the root, Executive Vice Presidents (EVPs) at the second level and so forth. Company organizational chart shows the company internal management structure as well as the relationships and relative ranks of employees with different positions/jobs, which is a common visual depiction of how a company is organized.

Nowadays, to facilitate the collaboration and communication among employees, a new type of online social networks named *enterprise social networks* (ESNs) has been adopted inside the firewalls of many corporations. A representative example of online ESNs is Yammer¹. Over 500,000 businesses around the world are now using Yammer, including 85% of the Fortune 500². Yammer provides employees with various enterprise social network services to help them deal with daily work issues and contains abundant heterogeneous information generated by employees’ online social activities.

Problem Studied: Company internal organizational chart is usually confidential to the public for the privacy and security reasons. In this paper, we want to infer the organi-

¹<https://www.yammer.com/>

²<https://about.yammer.com/why-yammer/>

zational chart of a company based on the heterogeneous information in online ESNs launched in the company, and the problem is formally named as the *Inference of Organization Chart* (IOC) problem.

To help illustrate the IOC problem more clearly, we also give an example in Figure 1, where the left plot is about an online ESN adopted in a company and the right plot shows the company’s organizational chart. In the ESN, users can have different types of social activities, e.g., follow other users, join groups, write/reply/like posts, etc. Meanwhile, in the organizational chart, employees are connected by supervision links from managers to subordinates, who are organized into a rooted tree of depth 2 with CEO “Adam” at the root. In companies, managers can usually supervise several subordinates simultaneously, while each subordinate only reports to one single manager. For instance, in Figure 1, CEO “Adam” manages “Bob” and “Candy” concurrently, while “David” only needs to report to “Bob”.

The IOC problem is an interesting yet important problem. Besides inferring company organizational chart, it can also be applied in other real-world concrete applications: (1) identifying the command structures of terrorist organizations [30] based on the communication/traffic networks of their members. The command structures of terrorist organizations are usually pyramid diagrams outlining their support systems consisting of the *leaders*, *operational cadre*, *active supporters* and *passive supporters* [30]. Uncovering their internal operational structure and determining roles of members will be helpful for conducting precise strikes against their key leaders and avoid the tragic events, like 9/11 [22]. (2) inferring the social hierarchies of animals based on their observed interaction networks [24]. Many animals (like, mammals, birds and insect species) are usually organized into dominance hierarchies. Identifying and understanding the organizational hierarchies of animals will be helpful to design and carry out effective conservation measures to protect them.

Albeit its importance, IOC is a novel problem and we are the first to propose to study it based on online ESNs. The IOC problem is totally different from existing works: (1) “*hierarchy detection in social networks*” [15], which only studies the division of the regular users of the social networks into different hierarchies, who are not actually involved in any organizations; (2) “*organizational intrusion*” [12], which focuses on attacking organizations and attaining company internal information only; and (3) “*inferring offline hierarchy from social networks*” [19], which merely infers fragments of offline hierarchical ties in *homogeneous* networks, instead of reconstructing the whole organizational chart. Different from all these works, in this paper, we aim at recovering the complete organizational chart of a company (including both the hierarchical tiers of employees and the supervision links from managers to subordinates) based on the *heterogeneous* information about the employees in online ESNs.

Meanwhile, to guarantee the smooth operations of companies, the inferred organizational chart needs to meet certain structural requirements [2], including both (1) macro-level depth requirement, and (2) micro-level width requirement. Two classical organizational structures adopted by companies are the vertical structure and the horizontal structure [6]. Vertical organizational structure with well-defined chains of command clearly outlines the responsibilities of each employee but will result in delays in information delivery [6].

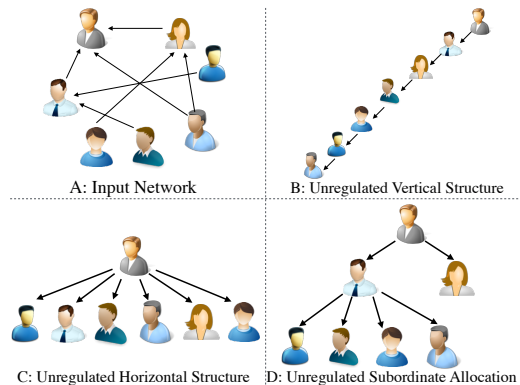


Figure 2: Examples of organizational charts.

Meanwhile, horizontal organizational structure with flat command system involves everyone in decision making but will lead to difficulties in coordinating the activities of different departments [6]. For instance, based on the input social network shown in plot A of Figure 2, we give two extreme cases of the vertical and horizontal organizational charts without depth regulation in plots B and C respectively, both of which will lead to serious management problems for large companies involving tens of thousands employees. Proper regulation of the inferred organizational chart’s depth (i.e., the macro-level depth requirement) is generally desired. On the other hand, most employees in companies need good supervisors to coach and instruct their daily work, but the number of subordinates each manager can supervise is limited, which can be determined by their management capacities, available time and energy. Rationally regulating the allocation of supervision workload among managers (i.e., the micro-level width requirement) can improve the management effectiveness significantly. For instance, in plot D of Figure 2, we show an inferred organizational chart with depth regulation but no subordinate allocation regulation. In the plot, users in ESN are stratified into 3 tiers (which is relatively reasonable compared to the extreme cases in plots B and C) but the employees’ management workloads at tier 3 are all assigned to one single manager, which may be beyond his/her management ability.

Despite its importance and novelty, the IOC problem is very hard to solve due to the following challenges:

- *regulated social stratification*: *Effective social stratification* to partition users into different hierarchical arrangements (i.e., identifying the relative manager-subordinate roles of employees) while meeting the macro-level depth requirement is the prerequisite for addressing the IOC problem.
- *supervision link inference*: Supervision link is a new type of link merely existing from managers to their subordinates. Predicting the existence of potential *supervision links* with the heterogeneous information in ESNs is still an open problem.
- *regulated supervision workload allocation*: To maximize the management effectiveness and efficiency, the number of subordinates each manager can supervise is limited by the management threshold K . In other words, supervision links in organizational chart have an inherent *K -to-one* constraint.

To address all the above challenges, a new unsupervised organizational chart inference framework named CREATE

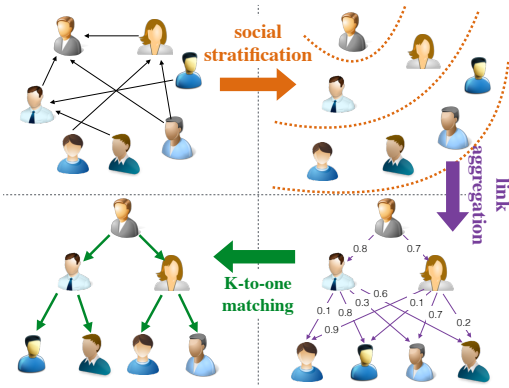


Figure 3: The framework of Create.

(ChArT REcovEr) is proposed in this paper. Several new concepts (e.g., “class transcendence social links”, “Matthew Effect based constraint” and “chart depth regulation constraint”) will be introduced and CREATE resolves the *regulated social stratification* challenge by minimizing the existence of class transcendence social links in ESNs. CREATE tackles the *supervision link inference* challenge by aggregating multiple social meta paths in the ESN between to consecutive social hierarchies. Finally, CREATE handles the *regulated supervision workload allocation* challenge by applying *network flow* to match consecutive social hierarchies to preserve the *K-to-one* constraint on *supervision links*.

The remaining parts of the paper are organized as follows. In Section 2, we will define some important terminologies and the IOC problem. Method CREATE will be introduced in Section 3. Extensive experiment results are available in Section 4. Finally, we describe the related works in Section 5 and conclude this paper in Section 6.

2. PROBLEM FORMULATION

In this section, we will introduce the formal definitions of “heterogeneous social network” and “organizational chart” at first and then define the IOC problem with these two concepts.

2.1 Terminology Definition

Definition 1 (Heterogeneous Social Networks): A *heterogeneous social network* can be represented as $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \bigcup_i \mathcal{V}_i$ and $\mathcal{E} = \bigcup_i \mathcal{E}_i$ are the sets of different types of nodes and complex links among these nodes in the network respectively.

As introduced in Section 1, users in online ESNs (e.g., Yammer) have various types of social activities, e.g., follow other users, join groups, write/reply/like online posts, etc. As a result, Yammer can be represented as a *heterogeneous social network* $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \mathcal{U} \cup \mathcal{G} \cup \mathcal{P}$ is the set of user, group and post nodes in G and $\mathcal{E} = \mathcal{S} \cup \mathcal{J} \cup \mathcal{W} \cup \mathcal{R} \cup \mathcal{K}$ denotes the set of social links among users, join links between users and groups, as well as write, reply and like links between users and posts respectively.

Definition 2 (Organizational Chart): The *organization chart* of a company can be represented as a *rooted tree* [8] $T = (\mathcal{N}, \mathcal{L}, root)$, where \mathcal{N} is the set of *employees*, \mathcal{L} denotes the set of directed *supervision links* from managers to subordinates in T and *root* represents the CEO in the company.

2.2 Problem Definition

Based on the definitions of *heterogeneous social network* and *organizational chart*, we can define the IOC problem formally as follows:

Definition 3 (Organizational Chart Inference (IOC)): Given an online ESN $G = (\mathcal{V}, \mathcal{E})$ launched in a company, the IOC problem aims at inferring the most likely organizational chart $T = (\mathcal{N}, \mathcal{L}, root)$ of the company, where $\mathcal{N} = \mathcal{U}$ (\mathcal{U} is the user set in network G). Furthermore, considering that the node set as well as the root node in T are fixed, the IOC problem actually aims at inferring the $|\mathcal{N} - 1|$ most likely *supervision links* \mathcal{L} among employees. The inferred supervision links together with the node set \mathcal{N} as well as the *root* node can recover the original organizational chart $T = (\mathcal{N}, \mathcal{L}, root)$ of the company.

3. PROPOSED METHODS

Considering that supervision links exist merely between managers and subordinates, we propose to stratify users in enterprise social networks into different *social classes* to identify their relative manager-subordinate roles in Subsection 3.1. Macro-level depth requirement of the inferred chart is achieved with the *depth regulation constraint* in the social stratification objective function. Potential supervision links can be inferred between employees in consecutive classes by aggregating social meta paths among employees in the ESN in Subsection 3.2. To preserve the *K-to-one* constraint on supervision links (i.e., the micro-level width requirement), redundant non-existing supervision links will be pruned in Subsection 3.3. Generally, as shown in Figure 3, framework CREATE has three steps: (1) regulated social stratification, (2) supervision link inference, and (3) regulated social class matching, which will all be introduced in this section.

3.1 Regulated Social Stratification

Supervision links merely exist between managers and subordinates. Division of users into hierarchies to identify their relative manager-subordinate roles can shrink the supervision link inference space greatly. The process of hierarchizing users in online ESN is called *social stratification* formally. **Definition 4** (Social Stratification): Traditional *social stratification* concept used in social science denotes the ranking and partition of people into different hierarchies based on various factors, e.g., power, wealth, knowledge and importance [32]. In this paper, we define *social stratification* as the partition process of users in online ESNs into different hierarchies according to their management relationships, where managers are at upper levels, while subordinates are at lower levels.

The relative stratified levels of users in online ESN are defined as their social classes.

Definition 5 (Social Class): *Social class* is a term used by social stratification models in social science and the most common ones are the upper, middle, and lower classes [13]. In this paper, we define *social class* of users in online ESNs as their management level in the company, where CEO belongs to social class 1, EVPs belong to class 2, and so forth.

In social stratification, users in online ESNs will be mapped to their *social classes* according to mapping: $c : \mathcal{U} \rightarrow \mathbb{Z}^+$. For each user $u \in \mathcal{U}$, his social class $c(u)$ is defined recur-

sively as follows:

$$c(u) = \begin{cases} 1, & \text{if } u \text{ is the CEO;} \\ c(m(u)) + 1, & \text{otherwise.} \end{cases}$$

where $m(u)$ represents the direct manager of u .

In social science, the working class are eager to get acquainted with and join the upper echelons of their class by either accumulating wealth [3], imitating their dressing styles [4], and mimicking their dialect and accents [23]. Meanwhile, the upper class are very cohesive and they tend to be friends who share similar background [17]. So is the case for the social links in *enterprise social networks*. By analyzing the Yammer network data, we observe that the probability for users to follow upper-level managers is 31.9% on average, while that of following subordinates is merely 11.2%. As a result, in online ESNs, subordinates tend to follow their managers, while people in management are reluctant to initiate the friendship with their subordinates [1]. Based on such an observation, we introduce the concept of *class transcendence* social links and propose to stratify users by minimizing the existence of such links in ESNs.

Definition 6 (Class Transcendence Social Link): Link (u, v) (i.e., u follows v) is defined as a *class transcendence social link* in online ESN G iff $(u, v) \in \mathcal{S}$ and $c(u) < c(v)$ (where smaller social class denotes upper management level in the organizational chart).

In social stratification, each introduced class transcendence social link in the result will lead to a *class transcendence penalty*. Let $c(\mathcal{U}) = \{c(u_1), c(u_2), \dots, c(u_{|\mathcal{U}|})\}$ be the *social stratification* result of all users in the ESN. For any directed social link $(u, v) \in \mathcal{S}$ in the ESN, the *class transcendence penalty* introduced by it can be represented as

$$p(c(u), c(v)) = \begin{cases} 0, & \text{if } c(u) > c(v) \\ c(v) - c(u) + 1, & \text{otherwise.} \end{cases}$$

The *class transcendence penalty* introduced by all social links (i.e., \mathcal{S}) in the ESN can be represented as

$$\begin{aligned} p(c(\mathcal{U})) &= \sum_{(u,v) \in \mathcal{S}} p(c(u), c(v)) \\ &= \sum_{(u,v) \in \mathcal{S}} \max\{c(v) - c(u) + 1, 0\}. \end{aligned}$$

“The rich get richer” (i.e., the Matthew Effect [25]) is a common phenomenon in social science literally referring to issues of fame or status as well as cumulative advantage of economic capital. By analyzing the Yammer network data, we have similar observations: “people at higher management level can accumulate more followers easily”. Such an observation provides important hints for inferring users’ relative management levels according to their in degrees in ESN (i.e., the number of followers).

Definition 7 (Matthew Effect based Constraint): For any two given users u and v in the network, let $\Gamma(u)$ and $\Gamma(v)$ be the follower sets of u and v in the network respectively. The *matthew effect based constraint* on users u and v can be represented as $c(u) \leq c(v)$ if $|\Gamma(u)| \geq |\Gamma(v)|$.

Furthermore, to maximize the operation efficiency of companies, the inferred organizational chart needs to meet the *macro-level depth requirement*, which can be achieved with the following chart depth regulation constraint.

Definition 8 (Chart Depth Regulation Constraint): The *chart depth regulation constraint* avoids obtaining organizational chart with too short command chains (e.g., the extreme horizontal structure) and can be represented as

$$\sum_{u \in \mathcal{U}} c(u) \geq \alpha \cdot |\mathcal{U}|,$$

where parameter α is used to regulate the depth of the chart, whose sensitivity analysis will be given in Section 4. Furthermore, term $\sum_{u \in \mathcal{U}} c(u)$ is also added to the minimization objective function to avoid obtaining charts with too long command chains (i.e., the extreme vertical structure).

Based on all the above remarks, the *optimal* regulated social stratification $c^*(\mathcal{U})$ of users in ESN can be obtained by solving the following objective function:

$$\begin{aligned} c^*(\mathcal{U}) &= \arg \min_{\{c(u_1), c(u_2), \dots, c(u_{|\mathcal{U}|})\}} \sum_{(u,v) \in \mathcal{S}} p(c(u), c(v)) + \sum_{u \in \mathcal{U}} c(u), \\ \text{s.t., } & p(c(u), c(v)) \geq c(v) - c(u) + 1, \forall (u, v) \in \mathcal{S}, \\ & p(c(u), c(v)) \geq 0, \forall (u, v) \in \mathcal{S}, \\ & c(u) \leq c(v), \forall u, v \in \mathcal{U}, \text{ if } |\Gamma(u)| \geq |\Gamma(v)|, \\ & \sum_{u \in \mathcal{U}} c(u) \geq \alpha \cdot |\mathcal{U}|, \\ & c(u) = 1, \text{ if } u \text{ is the CEO,} \\ & c(u) > 1, c(u) \in \mathbb{Z}^+, \forall u \in \mathcal{U} \setminus \{\text{CEO}\}, \\ & p(c(u), c(v)) \in \mathbb{Z}, \forall (u, v) \in \mathcal{S}. \end{aligned}$$

The integer programming objective function can be solved with open source toolkits, e.g., GLPK³, PuLP⁴, etc., very easily and the obtained results of variables $c(u_1), c(u_2), \dots, c(u_{|\mathcal{U}|})$ represent the inferred social classes of users in online ESN.

3.2 Supervision Link Inference with Social Meta Paths Aggregation

It is a challenge to estimate the supervision relations between the ESN members in consecutive social classes. Here we use the meta paths concept introduced in [28, 29, 40] to identify and evaluate different types relationship in ESN.

Social Meta Paths in Enterprise Social Networks

- *Follow*: User $\xrightarrow{\text{follow}}$ User, whose notation is “ $U \rightarrow U$ ” or $\Phi_1(U, U)$.
- *Follower of Follower*: User $\xrightarrow{\text{follow}}$ User $\xrightarrow{\text{follow}}$ User, whose notation is “ $U \rightarrow U \rightarrow U$ ” or $\Phi_2(U, U)$.
- *Common Followee*: User $\xrightarrow{\text{follow}}$ User $\xrightarrow{\text{follow}^{-1}}$ User, whose notation is “ $U \rightarrow U \leftarrow U$ ” or $\Phi_3(U, U)$.
- *Common Follower*: User $\xrightarrow{\text{follow}^{-1}}$ User $\xrightarrow{\text{follow}}$ User, whose notation is “ $U \leftarrow U \rightarrow U$ ” or $\Phi_4(U, U)$.
- *Common Group Membership*: User $\xrightarrow{\text{join}}$ Group $\xrightarrow{\text{join}^{-1}}$ User, whose notation is “ $U \rightarrow G \leftarrow U$ ” or $\Phi_5(U, U)$.
- *Reply Post*: User $\xrightarrow{\text{write}}$ Post $\xrightarrow{\text{reply}}$ Post $\xrightarrow{\text{write}^{-1}}$ User, whose notation is “ $U \rightarrow P \rightarrow P \leftarrow U$ ” or $\Phi_6(U, U)$.
- *Like Post*: User $\xrightarrow{\text{write}}$ Post $\xrightarrow{\text{like}^{-1}}$ User, whose notation is “ $U \rightarrow P \leftarrow U$ ” or $\Phi_7(U, U)$.

An existing user intimacy [37] measure, Path-Sim, based on meta paths was introduced in [29], which can calculate the propagation probability between users via meta paths

³<https://www.gnu.org/software/glpk/>

⁴<https://code.google.com/p/pulp-or/>

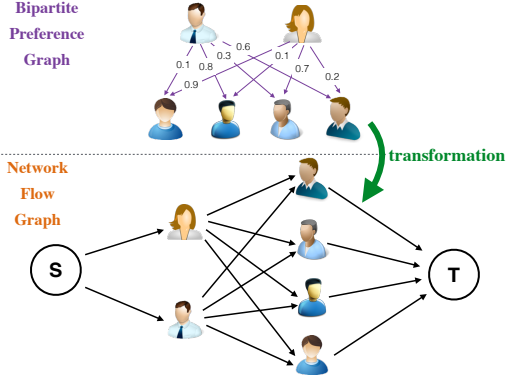


Figure 4: An example of K-to-one matching.

in *undirected homogeneous networks*. To deal with *directed heterogeneous networks*, we extend it and introduce a new intimacy measure, DP-intimacy (Directed Path-Intimacy), based on social meta path $\Phi_i(U, U)$, $i \in \{1, 2, \dots, 7\}$:

$$\text{DP-intimacy}_i(u, v) = \frac{|\mathcal{PATH}_i(u \rightsquigarrow v)| + |\mathcal{PATH}_i(v \rightsquigarrow u)|}{|\mathcal{PATH}_i(u \rightsquigarrow \cdot)| + |\mathcal{PATH}_i(v \rightsquigarrow \cdot)|},$$

where $\mathcal{PATH}_i(u \rightsquigarrow v)$ denotes the instance set of meta path $\Phi_i(U, U)$ going from u to v in the ESN.

Different social meta paths capture the intimacy between users in different aspects and overall intimacy between users can be obtained by aggregating information from all these social meta paths. Let DP-intimacy₁(u, v), DP-intimacy₂(u, v), \dots , DP-intimacy₇(u, v) be the intimacy scores between users u and v calculated based on social meta paths $\Phi_1(U, U)$, $\Phi_2(U, U)$, \dots , $\Phi_7(U, U)$ respectively. Without loss of generality, we choose logistic function as the intimacy aggregation function [14], the overall intimacy between users u and v can be represented as

$$\text{intimacy}(u, v) = \frac{e^{\sum_{(i)} \omega_i \text{DP-intimacy}_i(u, v)}}{1 + e^{\sum_{(i)} \omega_i \text{DP-intimacy}_i(u, v)}} \in [0, 1],$$

where the value of ω_i denotes the weight of social meta path Φ_i and $\sum_i \omega_i = 1$.

3.3 Regulated Social Class Matching

Meta path aggregation based supervision link inference method proposed in previous step calculate the intimacy scores of all the potential links between pairs of social classes. However, to regulate the supervision workload allocation, the number of subordinates each manager can supervise is limited by the management threshold K . In this section, we will prune the redundant non-existing supervision links with network-flow based regulated social class matching to preserve the *K-to-one* constraint on *supervision links*.

3.3.1 Bipartite Preference Graph

Based on the *social stratification* results $c(\mathcal{U}) = \{c(u_1), c(u_2), \dots, c(u_{|\mathcal{U}|})\}$, we can stratify all the users into *social classes* $[1, \max(c(\mathcal{U}))]$ and users in class $i \in [1, \max(c(\mathcal{U}))]$ can be represented as set $\Psi(i) \subset \mathcal{U}$. By aggregating information in various *social meta paths*, we can calculate the intimacy scores of all potential *supervision links* between consecutive *social classes*, which exist between $\Psi(i)$ and $\Psi(i+1)$ can be represented as set $\Lambda(i, i+1) = \Psi(i) \times \Psi(i+1)$. Links

in $\Lambda(i, i+1)$ are associated with certain weights (i.e., the calculated intimacy scores), which can be obtained with the mapping $\Pi(i, i+1) : \Lambda(i, i+1) \rightarrow \mathbb{R}$.

Users in social classes i and $i+1$ (i.e., $\Psi(i)$ and $\Psi(i+1)$) together with all the potential supervision links between them (i.e., $\Lambda(i, i+1)$) and their intimacy scores (i.e., $\Pi(i, i+1)$) can form a weighted bipartite preference graph.

Definition 9 (Weighted Bipartite Preference Graph): The *weighted bipartite preference graph* between users in $\Psi(i)$ and $\Psi(i+1)$ can be represented as $B = (\Psi(i) \cup \Psi(i+1), \Lambda(i, i+1), \Pi(i, i+1))$.

An example of *weighted bipartite preference graph* is shown in the upper plot of Figure 4. In the example, all the potential supervision links between the upper-level and lower-level individuals are represented as the directed purple lines between them, whose weights are the numbers marked on the lines. Each employee in the figure is associated with multiple potential *supervision links* and the redundant ones can be pruned with the *network flow* method introduced in the next subsection.

3.3.2 Minimum Cost Network Flow based Social Class Matching

Based on the *bipartite preference graph* B , we propose to construct the following *network flow graph* first.

Definition 10 (Network Flow Graph): Based on bipartite preference graph $B = (\Psi(i) \cup \Psi(i+1), \Lambda(i, i+1), \Pi(i, i+1))$, the *network flow graph* can be represented as $H = (\mathcal{N}_H, \mathcal{L}_H, \mathcal{W}_H)$. Node set \mathcal{N}_H includes all nodes in B and two dummy nodes: source node s and sink node t (i.e., $\mathcal{N}_H = \Psi(i) \cup \Psi(i+1) \cup \{s, t\}$). Besides all the links in B , we further add directed links from s to all nodes in $\Psi(i)$, as well as those from all nodes in $\Psi(i+1)$ to t (i.e., $\mathcal{L}_H = \Lambda(i, i+1) \cup (\{s\} \times \Psi(i)) \cup (\Psi(i+1) \times \{t\})$). Only the links in $\Lambda(i, i+1)$ are associated with weights, which can be obtained with mapping $\mathcal{W}_H = \Pi(i, i+1)$.

For instance, based on the *bipartite preference graph* in the upper plot of Figure 4, we can construct its corresponding *network flow graph* (i.e., the lower plot). All the links in the network flow graph are directed denoting the flow direction.

Bound Constraint of Network Flow

For each link $(u, v) \in \mathcal{L}_H$, we allow a certain amount of flow going through within range $[lb_{u,v}, ub_{u,v}]$, where $lb_{u,v}$ and $ub_{u,v}$ represent the *lower bound* and *upper bound* associated with link (u, v) respectively and

$$lb_{u,v} \leq x_{u,v} \leq ub_{u,v},$$

where $x_{u,v}$ is the flow amount going through link (u, v) .

More specifically, for links from s to the upper level individuals, i.e., $\{s\} \times \Psi(i)$, we set its *lower bound* and *upper bound* to be $lb_{s,u} = 0$ and $ub_{s,u} = K$ respectively and we can get

$$0 \leq x_{s,u} \leq K, \forall u \in \Psi(i),$$

where K is the management threshold, whose sensitivity analysis is available in Section 4. It is actually the constrain to preserve micro-level width requirement.

For link $(v, t) \in \mathcal{L}_H$, we set its *lower bound* and *upper bound* to be $lb_{v,t} = 1$ and $ub_{v,t} = 1$, i.e.,

$$x_{v,t} = 1, \forall v \in \Psi(i+1),$$

which means exact amount 1 flow goes through link (v, t) (i.e., each subordinate needs to have exactly one manager).

Links $(u, v) \in \Lambda(i, i + 1) \subset \mathcal{L}_H$ have lower and upper bounds $lb_{u,v} = 0$, $ub_{u,v} = 1$ and the flow amount needs to be an integer, i.e.,

$$x_{u,v} \in \{0, 1\},$$

denoting whether supervision links in $\Lambda(i, i + 1)$ are selected or not in the matching result.

Mass-Balance Constraint of Network Flow

In network flow model, for each node, e.g., u , the amount of flow going into u should be equal to that going out from u , i.e.,

$$\sum_{w \in \mathcal{N}_H, (w,u) \in \mathcal{L}_H} x_{w,u} = \sum_{v \in \mathcal{N}_H, (u,v) \in \mathcal{L}_H} x_{u,v}.$$

Minimum Cost Network Flow

All links going from $\Psi(i)$ to $\Psi(i + 1)$ are associated with corresponding *flow costs*, which are negatively correlated to their intimacy scores. For instance, in this paper, for link $(u, v) \in \mathcal{L}_B$ with weight $intimacy(u, v)$, we can represent their *flow cost* as $1 - intimacy(u, v)$. The optimal *network flow* with the *minimum cost* (i.e., the maximum intimacy) can be obtained by addressing the following *integer programming* problem:

$$\begin{aligned} \min \quad & \sum_{(u,v) \in \Lambda(i,i+1)} x_{u,v}(1 - intimacy(u,v)) \\ \text{s.t.} \quad & 0 \leq x_{s,u} \leq K, \text{ for } \forall u \in \Psi(i), \\ & x_{v,t} = 1, \text{ for } \forall v \in \Psi(i+1), \\ & x_{u,v} \in \{0, 1\}, \text{ for } \forall u \in \Psi(i), \forall v \in \Psi(i+1), \\ & \sum_{w \in \mathcal{N}_H, (w,u) \in \mathcal{L}_H} x_{w,u} = \sum_{v \in \mathcal{N}_H, (u,v) \in \mathcal{L}_H} x_{u,v}, \forall u \in \mathcal{N}_H. \end{aligned}$$

Similarly, the above integer programming problem can be addressed with open source toolkits and how to solve the equation will not be introduced here. Variables obtained by solving the above equation can lead to the minimum cost but can also meet the constraints as well. These obtained variables denote the existence scores of the corresponding supervision links, where the selected links (i.e., those corresponding variable $x = 1$) will be assigned with label +1 while the rest are assigned with label -1.

4. EXPERIMENTS

To examine the performance of CREATE in addressing the IOC problem, in this part, extensive experiments will be conducted on real-world *enterprise social network*: Yammer.

4.1 Dataset Description

We crawl all the Microsoft employees' information from Yammer and obtain the complete organizational chart involving all these employees in Microsoft during June, 2014. The social network data covers all the user-generated content (such as posts, replies, topics, etc.) and social graphs (such as user-user following links, user-group memberships, user-topic following links, etc.) by then that are set to be public. In summary, it includes more than 100k Microsoft employees, and millions of user-generated posts published and the social links.⁵

⁵We are not able to reveal the actual numbers here and throughout the paper for commercial reasons.

All the users in yammer are registered with the official employment ID in Microsoft, via which we can identify them in the organizational chart correspondingly. From Microsoft, the complete organization structure of all employees is obtained. As introduced before, the structure of the organizational chart is a rooted tree with the CEO at the top.

4.2 Experiment Settings

The CREATE framework proposed in this paper is an unsupervised model, and the *organizational chart* is used for evaluation only in the experiments. To ensure the employee node set of organizational chart to be identical to that of ESN, a fully aligned [21] Yammer network and organizational chart are sampled from the dataset. Initially, with the directed follow links among users in Yammer, we achieve the regulated social stratification of users by minimizing the number *class transcendence social links*. All the potential supervision links between pairwise consecutive social classes in the social stratification result are inferred by aggregating information from various social meta paths in Yammer, whose existence likelihood is denoted as the intimacy score. For simplicity, the weights of different social meta paths in logistic function are assigned with identical values, i.e., $\omega = [\frac{1}{7}, \dots, \frac{1}{7}]$. A subset of these inferred supervision links will be selected via the regulated social class matching based on the network-flow model to preserve the *K-to-one* constraint on supervision links.

Meanwhile, to demonstrate the effectiveness of CREATE, we compare CREATE with many baseline methods, including both state-of-art and traditional methods in social stratification and organizational chart inference.

Social Stratification Methods:

- *Regulated Social Stratification*: Regulated social stratification is the first step of CREATE proposed in this paper, which is also named as CREATE for simplicity. CREATE exploits the concept of class transcendence social links and Matthew Effect based constraint to stratify users in ESNs into different social classes. In addition, to regulate the depth of inferred social classes about employees, CREATE further adds a *chart depth regulation constraint* into the objective function.
- *Agony based Social Division*: ASD is a state-of-art social division method proposed in [15], which detects the social hierarchies of regular users in general online social networks. ASD is not designed for organizational chart inference and doesn't consider the matthew effect based constraint nor the chart depth regulation constraint.

Organizational Chart Inference Methods:

- *Social Stratification + Link Prediction + Matching* (CREATE): CREATE is the framework proposed in this paper and it has three steps: (1) regulated social stratification, (2) link inference and (3) regulated social class matching.
- *Social Stratification + Link Prediction* (CREATE-SL): CREATE-SL contains two steps: (1) *social stratification*, and (2) *link prediction* based on accumulated social meta paths. CREATE-SL has no *matching step* to keep the micro-level width requirement and the outputs cannot meet the *K-to-one* constraint.
- *Social Stratification + Matching* (CREATE-SM): CREATE-SM contains two steps: (1) *social stratification*, and (2) social class matching. CREATE-SM has no supervision

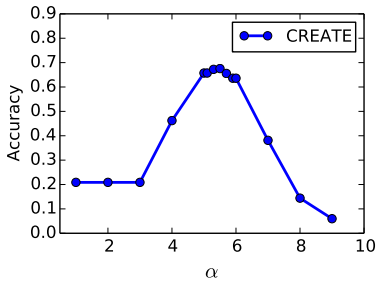


Figure 5: Sensitivity analysis of parameter α .

link prediction step and social links from upper-level social class to the lower-level are regarded as the potential supervision links candidates.

- *Social Stratification* (CREATE-S): CREATE-S is identical to CREATE-SL except that it has no matching step and outputs the all the *social links* between sequential hierarchies as the supervision links.
- *Traditional Unsupervised Link Prediction Methods*: No existing supervised link prediction models can be applied as no labeled supervision link exist. For completeness, we further compare *our* with traditional unsupervised baseline methods which include *Common Neighbor (CN)* [16], *Jaccard's Coefficient (JC)* [16] and *Adamic Adar (AA)* [5] between consecutive stratified social classes.

Social Stratification Evaluation Metrics

In the *social stratification* step, the outputs are the inferred *social classes* of all the employees. By comparing them with individuals' real-world *social classes* (i.e., the ground truth), we can calculate the *mean absolute error* [31], *mean squared error* [31] and *coefficient of determination* (i.e., R^2) [26] of the results. In addition, the ratio of correctly stratified users (i.e., accuracy) can also be used to measure the performance. So, the metrics used to evaluate the performance of different social stratification methods include *mean absolute error* (MAE) [18], *mean squared error* (MSE) [27], R^2 [27] and *accuracy*.

Organizational Chart Inference Evaluation Metrics

Methods CREATE-SL, CREATE-L and the traditional unsupervised link prediction can only output the confidence scores of all potential supervision links without labels, whose performance will be evaluated by metrics *AUC* and *Precision@100*. Meanwhile, CREATE and CREATE-SM can output both labels and scores of potential supervision links and, besides *AUC* and *Precision@100*, we will also evaluate their performance with *Precision*, *Recall* and *F1-score*.

4.3 Social Stratification Results

In social stratification, parameter α is applied to maintain the macro-level depth requirement, which can control the depth of the organizational chart. Before comparing the performance of CREATE with ASD, we will analyze the sensitivity of parameter α at first. We select α with values in $\{1, 2, 3, 4, 5, 5.1, 5.3, 5.5, 5.7, 5.9, 6, 7, 8, 9\}$ and obtain the accuracy scores achieved by CREATE as shown in Figure 5.

When parameter α is very small (e.g., from 1 to 3), we observe that it has no effects on the performance of CREATE. The possible reason can be that *Matthew Effect* based

constraint can already effectively outline the relative hierarchical relationships among users in online ESNs, the average social class of users obtained based on which is already greater than 3. When α becomes larger (from 4 to 6), the *structure regulation constraint* starts to matter more and the social stratification accuracy goes up steadily and achieve the highest value at 5.5, i.e., the default value of α in later experiments. CREATE performs better as α increases shows that the *structure regulation constraint* can stretch the organizational structure and stratify users in their correct social classes. However, as α further increases (i.e., from 6 to 9), the accuracy achieved by CREATE decreases dramatically. The reason can be that larger α stretches the organizational structure too much and put lots users into the wrong social classes. For example, it is nearly impossible for users to achieve 9 as the average social class, which is actually the largest social class in the sampled fully aligned organizational chart.

Social stratification results of CREATE and ASD are given in Figures 6-7, where Figure 6 shows the results achieved by CREATE and ASD at each social class (evaluated by precision and recall respectively) and Figure 7 shows their overall performance (evaluated by Accuracy, MAE, MSE and R^2).

From the microscopic perspective, we observe that CREATE performs better than ASD consistently at all social classes. CREATE achieves both 1.0 precision and 1.0 recall at social classes 1, 2, i.e., CREATE identifies the top 2 management levels of the company correctly. The performance of CREATE at other social classes is also very promising. For instance, the precision scores achieved by CREATE at social classes 3, 4, 7, 8 (besides 1, 2) are either 1.0 or close to 1.0 and the recall scores of CREATE at social classes 5, 6 are also very high, which all outperform those of ASD with significant advantages.

From a macroscopic perspective, the performance of CREATE in stratifying the whole user set in ESN is very excellent and much better than that of ASD. The Accuracy, MAE, MSE and R^2 scores achieved by CREATE are 0.68, 0.43, 0.71 and 0.37 respectively, which all outperforms those achieved by ASD. For example, the accuracy achieved CREATE is almost the triple of that obtained by ASD, while the MAE and MSE obtained by CREATE are merely the 28% and 19% of those achieved by ASD. In addition, ASD gets negative R^2 scores in identifying social classes of users in ESNs, which denotes that the identified users' social classes are massively disordered and have no linear correlation with the social class ground truth at all.

4.4 Organizational Chart Inference Results

CREATE has proved its excellent effectiveness in stratifying users in ESNs, based on which, we further study its performance in inferring the potential supervision links between pairs of consecutive social classes, whose results are evaluated by AUC, Precision@100 in Table 1 as well as by Recall, Precision and F1 in Figure 8.

In Table 1, we compare CREATE (of different management thresholds K) with all the other baseline methods, where CREATE (with parameter $K = 15$ and 20) performs the best. Compared with CREATE-SL (or CREATE-S), CREATE (or CREATE-SM) which has the matching step can identify supervision links more effectively. For instance, CREATE (with $K = 15$) outperforms CREATE-SL by over 20% in AUC and 6% in Precision@100, and CREATE-SM (with

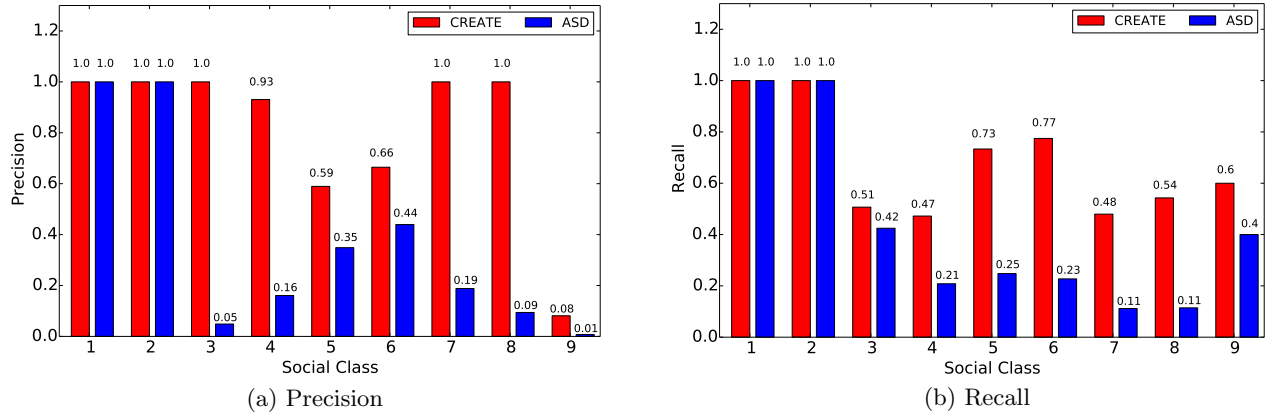


Figure 6: Precision and Recall achieved by Create and ASD at each social class of the organizational chart.

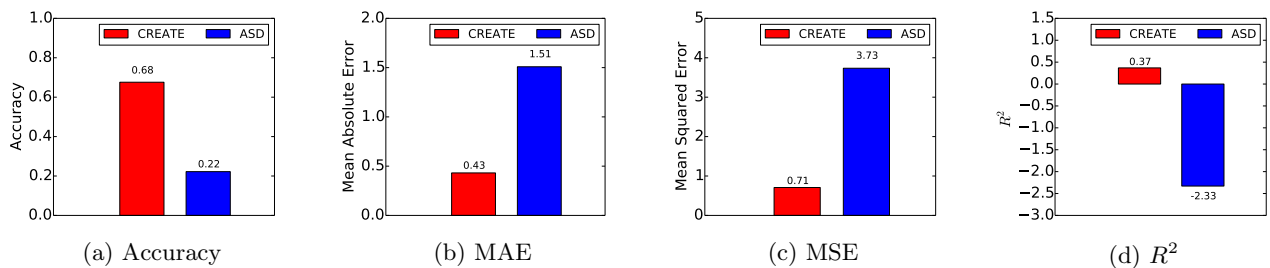


Figure 7: Performance comparison of Create and ASD evaluated by different metrics.

Table 1: Performance comparison of different organizational chart inference methods.

Method	Metrics	
	AUC	Precision@100
CREATE(K = 10)	0.856	0.830
CREATE(K = 15)	0.869	0.870
CREATE(K = 20)	0.869	0.870
CREATE-SL	0.719	0.820
CREATE-SM(K = 10)	0.610	0.720
CREATE-SM(K = 15)	0.630	0.790
CREATE-SM(K = 20)	0.630	0.790
CREATE-S	0.627	0.740
S-CN	0.636	0.440
S-JC	0.636	0.260
S-AA	0.528	0.070

parameter $K = 15$) outperforms CREATE-S with remarkable advantages. It demonstrates that matching step can effectively prune non-existing supervision links and preserve the micro-level width requirement (i.e., the K -to-one constraint).

Compared with CREATE-SM and CREATE-S, CREATE which infers the potential supervision links based on heterogeneous information in ESNs instead of merely regarding the social links as supervision link candidates achieves much better results. For example, in Table 1, the AUC of CREATE is 38% higher than that of CREATE-SM and CREATE-S, while the Precision@100 of CREATE is also roughly 10% higher as

well. In addition, in Figures 8, the Recall, Precision and F1 obtained by CREATE is almost triple of those achieved by CREATE-SM. It confirms the argument that heterogeneous information in ESNs can capture the relationships among colleagues (especially between managers and subordinates).

In addition, we also compare CREATE with traditional unsupervised link prediction methods, including CN, JC and AA, and the advantages of CREATE are very obvious according to Table 1: CREATE can outperform all these unsupervised link prediction methods with significant advantages.

4.5 Management Threshold Sensitivity Analysis

In social class matching, the management threshold parameter K plays a key role in constraining the number of supervision links connected to each managers. The sensitivity of parameter K will be analyzed in this section, where the results achieved by CREATE (with different K s) evaluated by different metrics are available in Figure 9. Small management threshold K (e.g., 5) limits each manager’s subordinate number to 5 and will preserve the supervision links with extremely high likelihood only but may miss many promising ones. However, as threshold K increases, more links with high likelihood will be preserved and the metric scores increase consistently. Meanwhile, when the threshold K goes to 25, the performance of CREATE degrades dramatically. The possible reason can be that, with larger threshold, each manager can have too many supervision links, which may exceed the subordinates they have in the real-world.

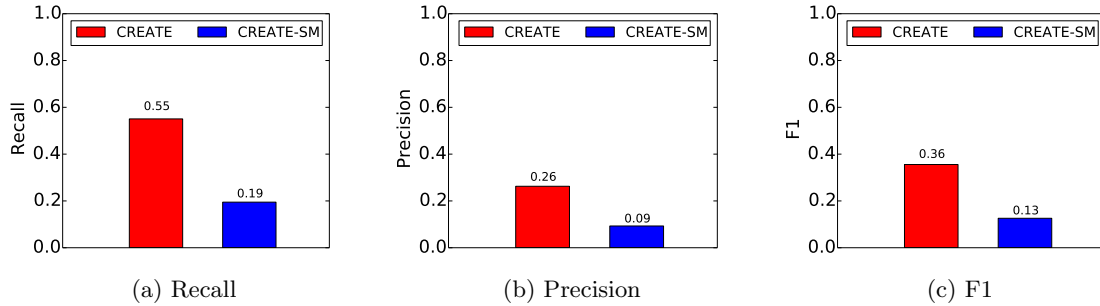


Figure 8: Performance comparison of Create and Create-sm evaluated by different metrics ($K = 15$).

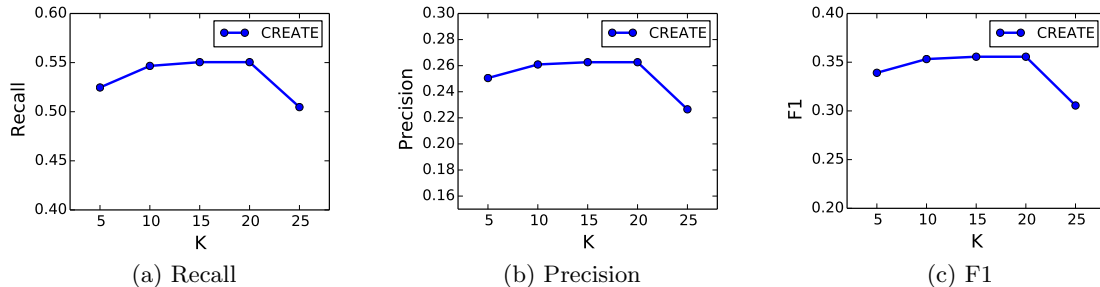


Figure 9: Sensitivity analysis of parameter K .

5. RELATED WORK

Enterprise social networks are important sources for employees in companies to get reliable information. Ehrlich et al. [11] propose to search for experts in enterprise with both text and social network analysis techniques. They propose to examine the users’ dynamic profile information and get the social distance to the expert before deciding how to initiate the contact. Enterprise social networks can lead to lots of benefits to companies and the motivations of enterprise social network adoption in companies are studied in details in [10]. Users in enterprise social networks will connect and learn from each other through personal and professional sharing. People sensemaking and relation building on an enterprise social network site is studied in by DiMicco et al. [9]. In addition, social connections among users in enterprise social networks usually have multiple facets. Wu et al. [33] propose to study the multiplexity of social connections among users in enterprise social networks, which include both professional and personal closeness.

From social networks, some works have been done to infer the hierarchies of individuals [7, 24, 15, 19]. A measure, agony, is proposed in [15], by minimizing which the authors propose a hierarchy detection method. A random graph model and markov chain monte carlo sampling is proposed by Clauset et al. in [7], which can address the problem of structural inference of hierarchies in networks. Maiya et al. propose to identify the hierarchies in social networks to achieve the maximum likelihood in [24]. All the above three papers focus on dividing individuals into different hierarchies only. A offline hierarchical ties inference method has been proposed by Jaber et al. in [19] to discover offline links among people based on a time-based model. However, none of these papers can recover the whole organizational chart.

Cross-social-network studies has become a hot research topic in recent years. Kong et al. [21] are the first to propose the concepts of “anchor links”, “anchor users”, “aligned networks” etc. A novel network anchoring method is proposed in [21] to address the network alignment problem. Cross-network heterogeneous link prediction problems are studied by Zhang et al. [35, 36, 40, 38] by transferring links across partially aligned networks. Besides link prediction problems, Jin et al. proposes to partition multiple large-scale social networks simultaneously in [20] and Zhang et al. study the community detection problem across partially aligned networks in [37, 39]. Zhan et al. analyze the information diffusion process across aligned networks [34].

6. CONCLUSION

In this paper, we have studied the organizational chart inference (IOC) problem based on the heterogeneous online ESNs. To address the IOC problem, a new chart inference framework CREATE has been proposed in Section 3. CREATE consists of 3 steps: (1) regulated social stratification, (2) supervision link inference with social meta paths aggregation, and (3) regulated social class matching. Experiments on real-world ESN and organizational chart dataset have demonstrated the effectiveness of CREATE.

7. ACKNOWLEDGEMENT

This work is supported in part by NSF through grants CNS-1115234, Google Research Award, the Pinnacle Lab at Singapore Management University, and Huawei grants.

8. REFERENCES

- [1] Be the boss, not a friend.
<http://www.pereg.com/manager/FCKeditor/editor/>

- filemanager/connectors.aspx/pereg.com/userfiles/file/Be%20the%20boss.pdf. [Online; accessed 24-December-2014].
- [2] Organisation-organizational structure-organisational chart. <http://kalyan-city.blogspot.com/2010/06/organisation-organizational-structure.html>. [Online; accessed 29-January-2015].
- [3] Rich, poor, and middle class life. http://www.historydoctor.net/Advanced%20Placement%20European%20History/Notes/rich_poor_and_middle_class_life.htm. [Online; accessed 24-December-2014].
- [4] The social foundations of fashion change in the late 19th century. <http://www.marquise.de/en/1800/ch1800soz.shtml>. [Online; accessed 24-December-2014].
- [5] L. Adamic and E. Adar. Friends and neighbors on the web. *Social Networks*, 2001.
- [6] M. Aoki. Horizontal vs. vertical information structure of the firm. *The American Economic Review*, 2009.
- [7] A. Clauset, C. Moore, and M. Newman. Structural inference of hierarchies in networks. In *Statistical Network Analysis: Models, Issues, and New Directions*. 2007.
- [8] R. Diestel. *Graph Theory*. Springer, 1997.
- [9] J. DiMicco, W. Geyer, D. Millen, C. Dugan, and B. Brownholtz. People sensemaking and relationship building on an enterprise social network site. In *HICSS*, 2009.
- [10] J. DiMicco, D. Millen, W. Geyer, C. Dugan, B. Brownholtz, and M. Muller. Motivations for social networking at work. In *CSCW*, 2008.
- [11] K. Ehrlich, C. Lin, and V. Griffiths-Fisher. Searching for experts in the enterprise: Combining text and social network analysis. In *GROUPE*, 2007.
- [12] A. Elishar, M. Fire, D. Kagan, and Y. Elovici. Organizational intrusion: Organization mining using socialbots. In *SocialInformatics*, 2012.
- [13] J. Grant. Class, definition of. In R.J. Barry Jones, editor, *Routledge Encyclopedia of International Political Economy*. 2001.
- [14] H. Gui, Y. Sun, J. Han, and G. Brova. Modeling topic diffusion in multi-relational bibliographic information networks. In *CIKM*, 2014.
- [15] M. Gupte, P. Shankar, J. Li, S. Muthukrishnan, and L. Iftode. Finding hierarchy in directed online social networks. In *WWW*, 2011.
- [16] M. Hasan and M. J. Zaki. A survey of link prediction in social networks. In Charu C. Aggarwal, editor, *Social Network Data Analytics*. 2011.
- [17] S. Hill. Elite and upper-class families. In *Families: A Social Class Perspective*. 2012.
- [18] R. Hyndman and A. Koehler. Another look at measures of forecast accuracy. *IJF*, 2006.
- [19] M. Jaber, P. Wood, P. Papapetrou, and S. Helmer. Inferring offline hierarchical ties from online social networks. In *WWW Companion*, 2014.
- [20] S. Jin, J. Zhang, P. Yu, S. Yang, and A. Li. Synergistic partitioning in multiple large scale social networks. In *IEEE BigData*, 2014.
- [21] X. Kong, J. Zhang, and P. Yu. Inferring anchor links across heterogeneous social networks. In *CIKM*, 2013.
- [22] V. Krebs. Mapping networks of terrorist cells. *Connections*, 2002.
- [23] A. Kroch. Dialect and style in the speech of upper class philadelphia. In *Towards a Social Science of Language: Papers in honor of William Labov*. 1995.
- [24] A. Maiya and T. Berger-Wolf. Inferring the maximum likelihood hierarchy in social networks. In *CSE*, 2009.
- [25] R. Merton. The matthew effect in science, ii: Cumulative advantage and the symbolism of intellectual property. *ISIS*, 1988.
- [26] N. Nagappan, T. Ball, and A. Zeller. Mining metrics to predict component failures. In *ICSE*, 2006.
- [27] J. Rawlings, S. Pantula, and D. Dickey. *Applied Regression Analysis*. 1988.
- [28] Y. Sun, R. Barber, M. Gupta, C. Aggarwal, and J. Han. Co-author relationship prediction in heterogeneous bibliographic networks. In *ASONAM*, 2011.
- [29] Y. Sun, J. Han, X. Yan, P. Yu, and T. Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. In *VLDB*, 2011.
- [30] US Army Training and Doctrine Command. *Terrorist Organizational Models*. 2007.
- [31] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE IP*, 2004.
- [32] L. Warwick-Booth. *Social Inequality: A Student's Guide*. SAGE Publications Ltd, 2013.
- [33] A. Wu, J. DiMicco, and D. Millen. Detecting professional versus personal closeness using an enterprise social network site. In *CHI*, 2010.
- [34] Q. Zhan, S. Wang, J. Zhang, P. Yu, and J. Xie. Influence maximization across partially aligned heterogeneous social networks. In *PAKDD*, 2015.
- [35] J. Zhang, X. Kong, and P. Yu. Predicting social links for new users across aligned heterogeneous social networks. In *ICDM*, 2013.
- [36] J. Zhang, X. Kong, and P. Yu. Transferring heterogeneous links across location-based social networks. In *WSDM*, 2014.
- [37] J. Zhang and P. Yu. Community detection for emerging networks. In *SDM*, 2015.
- [38] J. Zhang and P. Yu. Integrated anchor and social link predictions across partially aligned social networks. In *IJCAI*, 2015.
- [39] J. Zhang and P. Yu. Mcd: Mutual clustering across multiple heterogeneous networks. In *IEEE BigData Congress*, 2015.
- [40] J. Zhang, P. Yu, and Z. Zhou. Meta-path based multi-network collective link prediction. In *KDD*, 2014.