# Inferring Offline Hierarchical Ties from Online Social Networks

Mohammad Jaber and Peter T. Wood
Dept. of Computer Science and Information Systems
Birkbeck, University of London
London, UK

Panagiotis Papapetrou
Dept. of Computer and Systems Sciences
Stockholm University
Stockholm, Sweden

Sven Helmer
Faculty of Computer Science
Free University of Bozen-Bolzano
Bolzano, Italy

## ABSTRACT

Social networks can represent many different types of relationships between actors, some explicit and some implicit. For example, email communications between users may be represented explicitly in a network, while managerial relationships may not. In this paper we focus on analyzing explicit interactions among actors in order to detect hierarchical social relationships that may be implicit. We start by employing three well-known ranking-based methods, PageRank, Degree Centrality, and Rooted-PageRank (RPR) to infer such implicit relationships from interactions between actors. Then we propose two novel approaches which take into account the time-dimension of interactions in the process of detecting hierarchical ties. We experiment on two datasets, the Enron email dataset to infer manager-subordinate relationships from email exchanges, and a scientific publication co-authorship dataset to detect PhD advisor-advisee relationships from paper co-authorships. Our experiments show that time-based methods perform considerably better than ranking-based methods. In the Enron dataset, they detect 48% of manager-subordinate ties versus 32% found by Rooted-PageRank. Similarly, in co-author dataset, they detect 62% of advisor-advisee ties compared to only 39% by Rooted-PageRank.

## 1. INTRODUCTION

Social networks have been attracting the attention of the data mining community over the past decade. One task of particular interest is that of inferring social ties among members of a social network. In many cases such ties are explicit, e.g., a person identifies his/her friends or relatives. Well-known examples of such networks are Facebook, where people explicitly define their friendship relations, and Twitter, where users explicitly indicate whom they follow.

However, social ties may not always be explicit. In some applications, the online interactions between a group of people form a network that can be analyzed further to infer offline social relations between the group members. Consider, e.g., the network of emails exchanged between employees within a company (see Figure 1). By analyzing this network, we may be able to infer the direct manager of each employee. As another example, in a co-authorship network, explicit relationships exist between co-authors, where authors are linked to all their co-authors. Analyzing these explicit relationships may help detect other interesting social ties such as which pairs of co-authors have a PhD advisor-advisee relationship.

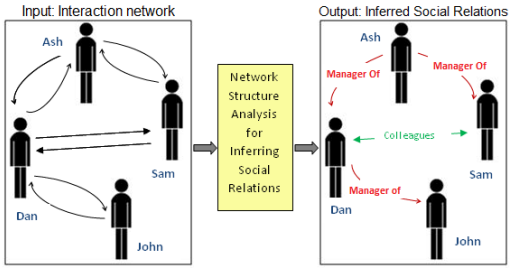


**Figure 1: Inferring implicit social relationships from interaction networks.**

In this paper, we are interested in the following problem: given a set of actors who are connected via a social network, infer potential hierarchical ties that may exist between these actors. For example, given a set of employees in a company, we are interested in ties such as manager-subordinate, or, given a set of co-authors, we are interested in ties such as advisor-advisee. Our proposed approach takes into account the online interaction between the actors, and infers these hierarchical ties by exploiting interaction patterns that occur during their communication. Moreover, we investigate the temporal dimension of user interactions. Our intuition is that actors connected by a hierarchical tie will exhibit different temporal interaction patterns to those who are connected by some other type of tie. Our findings demonstrate that for our problem setting "time matters".

Detecting hierarchical social ties may be beneficial in many ways and for different application domains. Firstly, they can be used for classifying actors according to their role in an organization or discovering communities and analyzing inter or intra-community relations. Secondly, hierarchical social ties may be used for detecting influential actors within a social network and studying how they affect or influence the whole network. Finally, such ties can be used for validating the influence attribution theory in social science, where


*WWW'14 Companion,* April 7–11, 2014, Seoul, Korea.
ACM 978-1-4503-2745-9/14/04.
http://dx.doi.org/10.1145/2567948.2579694.

it is assumed that information usually flows from "opinion leaders" to "ordinary users"[14].

The main contributions of this paper are as follows:

- We study the problem of inferring hierarchical ties in a social network, approaching it as a ranking problem.
- We employ three standard link-analysis ranking methods, vertex-degree centrality, PageRank, and Rooted-PageRank, as baseline approaches.
- We propose two novel time-based approaches, called Time function (Time-F) and filter-and-refine (FiRe), which take into account the time dimension of the interactions.
- We study the performance of these methods in terms of recall on two large real social networks: the Enron e-mail network and a co-authorship network. Our experiments show that the time-based methods achieve considerably better results than Rooted-PageRank (RPR). In the Enron network, up to 48% of manager-subordinate ties were detected by time-based methods, compared to only 32% by RPR. Similarly in co-author network, about 62% of advisor-advisee ties were detected by time-based methods, a significant improvement over the 39% achieved by RPR.

## 2. RELATED WORK

Few researchers have focused on finding implicit ties in social networks. One of the studies that we rely on is by Tang et al. [14] who incorporated four basic social psychological theories into a machine learning model. They generalised the problem of missing social relations by inferring these ties over multiple heterogeneous networks. In our study, we reuse the opinion leader theory they employed and find the appropriate settings for link analysis techniques to obtain the best results. Gupte at al. [4] proposed an algorithm to find the best hierarchy in a directed network. However, they studied the problem on a network-level, whereas our approach processes each actor individually. Buke et al. [3] focused on child-parent relationships at many life stages and how communication varies with the age of child, geographic distance and gender. Differently from our approach, they model the language used between users to generate text features. Backstrom et al. [1] developed a new measure of tie strength which they termed "dispersion" to infer romantic and spouse relationships. However, dispersion does not seem relevant to our problem of detecting hierarchical relations.

On the other hand, a lot of effort has been spent in the area of estimating a user's influence in a network. Li et al. [15] proposed a novel model based on the PageRank algorithm [2]. To evaluate the user's influence, they considered three factors: the number of the user's friends, the quality of his/her friends and the community label, i.e. the similarity between the user and the community. Many methods have been developed in social-network analysis to assess the importance of individuals in implicitly- or explicitly-defined social networks. In these cases, the network is represented as a directed graph, and the concept of *in-degree* (the number of incoming edges at a node), or refinements [6], is the simplest measure of the importance of a node. Other notions of importance in social networks include degree centrality, closeness centrality, betweenness centrality and eigenvector centrality [16, 12, 7, 13]. Liebowitz [11] applied the analytic hierarchy process to determine the strength of connections between actors.

A prominent application domain for measures of importance is in the area of web ranking. The two best-known techniques are PageRank [2] and HITS [8]. Many variants of these methods have been proposed, as well as adaptations of them for different objectives.

More recently, the temporal dimension of the ranking problem has been studied. Li et al. [10] investigated how time-based features improve the results of retrieving the relevant research publications in search engines. In their approach, not only the structure of the citation network is important but also the date of publication, i.e., older papers are given lower weight, so more recent results are ranked higher. Huang and Lin [5] implemented an approach that considers the temporal evolution of link occurrences within a social network to predict link occurrence probabilities at a particular time in the future. In our study we adopt a similar approach that uses link-based algorithms with temporal-based algorithms to achieve better results. However, our approach differs in that it detects *hierarchical* social ties.

## 3. BACKGROUND

We represent a social network as a graph $G = (V, E^c, E^s, \Theta)$ where:

- $V = \{v_1, \ldots v_n\}$ is the set of actors in the social network. We often refer to actors by $u$ and $v$.
- $E^c$ is the set of edges representing the interactions between the network actors. For example, in the case of an e-mail network, this represents e-mails exchanged between actors. Function $\Theta$ maps an interaction edge to the pair of actors involved:

  $$\Theta : E^c \rightarrow V \times V$$

  This allows for multiple interactions between a pair of actors.
- $E^s \subseteq V \times V$ is a set of pairs of actors, representing the hierarchical relationship between the actors. If $(u, v) \in E^s$, then $u$ is the direct superior of $v$ in the hierarchy (which we assume forms a forest). In the context of an e-mail network among a group of employees, $E^s$ might represent a manager-subordinate relationship.

The problem we wish to solve is as follows: given the subgraph $G_{input}(V, E^c, \Theta)$ of $G$ representing interactions among actors, we want to infer the subgraph of $G$ representing hierarchical social ties between actors $G_{output}(V, E^s)$. For example, given a set of e-mails exchanged among employees, or papers co-authored by authors, we want to infer manager-subordinate ties in a company, or advisor-advisee ties in academia, respectively.

## 4. RANKING-BASED METHODS

In this section, we present a baseline scheme based on two ranking metrics: degree centrality and PageRank. Then we consider a method based on Rooted PageRank. The key idea of these approaches is based on "opinion leader" theory [9]. The theory assumes that ideas/innovation usually flow first to opinion leaders, and from them to all other actors in a

network. In other words, actors who are classified as opinion leaders have more influence on others than ordinary actors. Accordingly, opinion leaders are more likely to have higher social position (such as managers or advisors) than ordinary actors (such as subordinates and advisees).

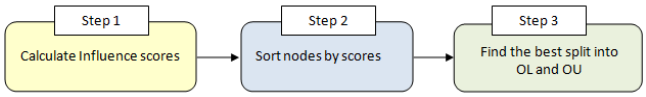


**Figure 2: Main steps in the baseline approach.**

## 4.1 The Baseline Scheme

As a baseline, we classify actors in the network into two classes: (a) opinion leaders ($OL$) and (b) ordinary users ($OU$). To do this, we follow three steps (see Figure 2):

1. First we assign a score to each actor. This score should reflect the importance (influence) of the actor in the whole network by analyzing the interaction network structure. We employ PageRank (PR) and Degree Centrality (DC) to assign a score to each node, as defined by the following two functions:

$$Score_{PR} : V \to \mathbb{R} \text{ and } Score_{DC} : V \to \mathbb{R}$$

2. We rank the actors in descending order of their scores. The expectation is that nodes in $OL$ should be ranked higher than those in $OU$.

3. Finally, assuming that the set of hierarchical relationships $E^s$ is known, we find the best position to split the sorted list of actors into two sub-lists, one representing $OL$ and the other $OU$. The best position is the one that maximizes the percentage of hierarchical relationships $(x, y) \in E^s$ ($x$ is the superior of $y$) for which it is the case that $x \in OL$ and $y \in OU$.

## 4.2 Rooted PageRank-based Approach

The above baseline scheme attempts to infer hierarchical relationships based on the *global* importance of each actor. However, a hierarchical relationship is specific to an individual actor: actor $x$ may have a big influence on actor $y$ but have no effect on actor $z$. Hence, we want to evaluate the importance of actors relative to a given actor.

To do so, we employ Rooted-PageRank (RPR). RPR calculates the importance scores of nodes relative to the root nodes it is given as input. In our application, RPR is given a single root node representing the actor whose hierarchical relationship we are trying to detect. For example, in the e-mail dataset, a manager $a$ should be important to his/her subordinate $b$. As a result, $a$ should have high rank among the most influential actors for $b$. Once again, this approach has three steps:

1. For each actor $x$ we run RPR with root node $x$ to evaluate the importance of nodes relative to $x$. Each node $y$ ($y \neq x$) in the network will have a score given by the function $RS_x(y) = RPR_x(y)$.

2. Then for each $x$ we produce a sorted list $L(x) = [y_1, y_2, \ldots, y_i, \ldots, y_{|V|-1}]$, such that $RS_x(y_i) \geq RS_x(y_{i+1})$ and $1 \leq i \leq n-1$.

3. Finally, assuming the hierarchical relationships $E^s$ are known and $(x, y_i) \in E^s$, the position of $y_i$ in $L(x)$ is given by the function $pos(L(x), y_i) = i$.

In order to evaluate this approach we define a function $\rho$ on positions, such that, for position $i$, $\rho$ gives the percentage of nodes whose direct superior in the hierarchical relationship appears within the top $i$ positions in their ranking of influential nodes. Function $\rho$ is defined as follows:

$$\rho(i) = \frac{|U|}{|W|} * 100 \quad (1)$$

where $W = \{x \mid x \in V \wedge (x, y) \in E^s\}$, and $U = \{x \mid x \in W \wedge pos(L(x), y) \leq i\}$. Here $|W|$ represents the total number of hierarchical ties. The results of the evaluation are discussed in Section 6.

# 5. TIME-BASED METHODS

In this section, we propose methods which consider the time dimension of actor interactions, in an attempt to improve on the results of Rooted-PageRank. For each pair of actors we define a time-series (based on interaction timestamps) representing their interaction over time. Analyzing the time-series patterns of such interactions could improve the detection of hierarchical ties, in that two actors who are related by a hierarchical tie may interact over time in a different way from how they interact with other actors.

## 5.1 Time Function Model (Time-F)

As in Rooted-PageRank, for each actor $x$, we rank all other actors according to their calculated scores. For each actor $x$, we find the rank of the actor $y$ who is directly connected to $x$ by a hierarchical tie. However, in this model, the scores employed to rank actors are calculated as follows:

$$TS_x(y) = \sum_{t=1}^{n} f_{xy}(t) \quad (2)$$

where:

- $x$ is the target actor.
- $TS_x(y)$ is the score of actor $y$ with respect to $x$
- $t$ is a time slot.
- $n$ is the total number of time slots.
- $f_{xy}(t)$ is the score between $x$ and $y$ over time slot $t$.

The definition of $f_{xy}(t)$ varies according to the meaning of hierarchical ties we try to detect. For example, if we are trying to detect a hierarchical tie which might be indicated by frequent and regular interactions, we define $f_{xy}(t)$ as follows:

$$f_{xy}(t) = \begin{cases} \frac{n_t}{N_t} & \text{if } N_t > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where

- $n_t$ is the total number of interactions between $x$ and $y$ within $t$.
- $N_t$ is the total number of interactions between actor $x$ and all other actors within $t$.

On the other hand, if we expect a hierarchical tie to be indicated by early interactions, we define $f_{xy}(t)$ as follows:

$$f_{xy}(t) = \begin{cases} (1 - \frac{n_s}{N_s}) \times \frac{n_t}{N_t} & \text{if } N_t, N_s > 0 \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

where:

- $n_s$ is the number of time slots between $t$ and the first interaction by $x$.
- $N_s$ is the number of slots between the first and last interactions by $x$.
- $n_t$ and $N_t$ are as above.

In this function, $y$ will have a high score in $x$'s ranked list if $x$ and $y$ interacted in early stages of $x$'s activities. This is captured by the term $(1 - \frac{n_s}{N_s})$ in the formula above. Moreover, in those early stages, the proportion of $x$'s interactions with his/her superior $y$ is supposed to be greater than the proportion of $x$'s interactions with others. This is captured by the term $\frac{n_t}{N_t}$ in Equation (4).

We use definition (3) for detecting manager-subordinate ties where interactions are emails exchanges, and definition (4) for detecting advisor-advisee ties where interactions are paper co-authorships which are generally more intensive in the early stages of the advisee's activities.

## 5.2 Filter-and-Refine Model (FiRe)

Here, we *filter* using Time-F and then *refine* using RPR. The process of detecting hierarchical ties for an actor $x$ consists of four steps:

1. Order the list of actors by the time-series function scores ($TS_x(y)$) and Rooted-PageRank scores ($RS_x(y)$), generating the ranked lists $LT_x = [a_1, a_2, \ldots, a_k, \ldots, a_{|V|-1}]$ and $LR_x = [b_1, b_2, \ldots, b_k, \ldots, b_{|V|-1}]$, respectively, where $TS_x(a_i) \geq TS_x(a_{i+1})$, $i = 1, 2, \ldots, |V| - 2$, and $RS_x(b_i) \geq RS_x(b_{i+1})$, $i = 1, 2, \ldots, |V| - 2$.

2. "Filtering step": Truncate list $LT_x$ at position $k$ to generate list $LT_x(k)$. If the truncation position $k$ is within a group of actors who have equal scores, we include all these actors in the filtered list $LT_x(k)$. Hence, we have:

$$LT_x(k) = \begin{cases} [a_1, \ldots, a_k], \text{ if } TS_x(a_k) > TS_x(a_{k+1}) \\ [a_1, \ldots, a_{k+s}], \text{ if } TS_x(a_i) = TS_x(a_{i+1}), \\ i = k, \ldots, k + s - 1 \text{ and} \\ TS_x(a_{k+s}) > TS_x(a_{k+s+1}) \end{cases} \quad (5)$$

3. "Refine step": Re-order $LT_x(k)$ (obtained in step 2) according to the Rooted-PageRank scores (obtained in step 1). This results in the list $LTR_x = [a_{\pi(1)}, \ldots, a_{\pi(k)}]$ where $\pi$ defines a permutation of $1, \ldots, k$ and $RS_x(a_{\pi(i)}) \geq RS_x(a_{\pi(i+1)})$, $i = 1, \ldots, k - 1$.

4. "Detecting the Relations": From the ranked list $LTR_x$ obtained in step 3, we detect the position $i$ of actor $y$, who is related to $x$ by a hierarchical tie, i.e.,

$$Pos(LTR_x, y) = \begin{cases} i & \text{if } LTR_x[i] = y \\ n & \text{if } y \notin LTR_x \end{cases} \quad (6)$$

| | Enron | co-author |
|---|---|---|
| $\|V\|$ | 155 | 1036990 |
| $E^c$ type | Emails | co-authoring |
| $\|E^c\|$ | 255632 | 1632442 |
| $E^s$ type | manager-subordinate | advisor-advisee |
| $\|E^s\|$ | 147 | 2098 |

**Table 1: Statistics of Enron and co-author datasets.**

Intuitively, $Pos(LTR_x, y)$ represents the number of actors who are ranked at least as high as the correct superior of $x$. In the ideal case $y$ should come first in $LTR_x$ ($Pos(LTR_x, y) = 1$). In the worst case, $y$ may be filtered out from the list during the filtering step. In such case, we have to consider the full set of actors as candidates for the direct superior position.

We use the same function as defined by formula (1) in order to evaluate this approach.

# 6. RESULTS AND ANALYSIS

In this section, we first describe the datasets we used for evaluation. Next we explain the experimental setup for the prediction process. Then we consider the baseline results we obtained for both PageRank (PR) and Degree Centrality (DC). After that, we show how rooted PageRank improves the detection of hierarchical relationships over both baseline approaches. Finally we present the significant improvement we obtained using Time-F and FiRe respectively.

## 6.1 Datasets

We evaluated the methods in terms of recall on two different datasets, Enron and co-author, both of which are available online[1]. Table 1 reports some statistics for each dataset.

The Enron dataset includes more than 255000 emails exchanged among 87474 email addresses between January 2000 and November 2001. However, only 155 of these email addresses belong to Enron employees. Each email in the dataset has a sender, subject, time stamp, body, and a set of receivers. The dataset also contains the hierarchical manager-subordinate relationship between employees. The co-author dataset includes more than 1 million authors who contributed to about 80000 papers in total between 1935 and 2011. Each paper has a title, date, conference where it was published and a list of co-authors. The dataset includes the hierarchical advisee-advisor relationship between the authors.

## 6.2 Experimental Setup

Before running the experiments, we had to decide exactly which features to include in the graph for each dataset.

### 6.2.1 Enron

We excluded all nodes (email-addresses) belonging to people not employed by Enron, since they would only add noise to our analysis. As a result, only emails exchanged between Enron employees were considered. Moreover, we ran the experiments on the unweighted directed graph. In other words, if person $u$ sent $n$ emails to person $v$, this is reflected in the graph as one edge $e$ where $\Theta(e) = (u, v)$. For Time-F and FiRe, each time-slot represents one week.

[1]http://arnetminer.org/socialtie/

### 6.2.2 Co-author

Since the co-author relationship is symmetric, the graph representing this dataset is undirected. Like the Enron dataset, the graph is unweighted: there is only one edge between a pair of authors when they coauthor at least one paper. For Time-F and FiRe, each time-slot represents one year. Due to the large size of the co-author dataset, we used *htcondor*[2] with 150 nodes in order to run the experiments, which still took 3.5 hours to complete in the worst case.

## 6.3 Baseline Results

As mentioned earlier, we want to find the position at which to split the list of actors into sublists $OL$ (opinion leaders) and $OU$ (ordinary users) such that the maximum percentage of hierarchical relationships $(u, v)$ have $u \in OL$ and $v \in OU$. Figure 3(a) shows the results obtained for the Enron dataset. For PageRank (PR), the best split position is when we use as $OL$ the top 20% of the sorted list of employees. At that position, for 72% of manager-subordinate relationships, the manager appears in $OL$ and the subordinate in $OU$. Degree Centrality (DC) reveals the same pattern but with worse results: the best split position is after the top 35% of the sorted list, with only 56% of manager-subordinate relationships correctly classified as $OL$-$OU$.

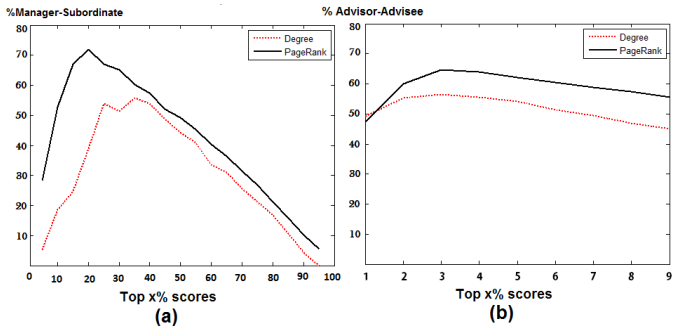


**Figure 3: Results using PageRank (PR) and Degree Centrality (DC) on (a) the Enron dataset and (b) the co-author dataset.**

In the co-author dataset, the best split position is roughly the same for both PR and DC, namely after the top 3% (see Figure 3(b)), with 65% and 56% of the advisor-advisee relationships being correctly classified, respectively.

## 6.4 Rooted PageRank (RPR) Results

We ran Rooted-PageRank (R-PR) $n$ times, where $n$ is the number of actors in the network, with a different actor as the root node each time. Figure 4(a) shows the results for R-PR on the Enron dataset. R-PR returns the manager as the top employee in the ranked list for about 32% of the subordinates. Moreover, for about 90% of subordinates , the manager is in the top 5.1% (top 8 employees) of their ranked list of employees. This percentage increases to 95% when we consider the top 6.4% of the results (top 10 employees) in each subordinate's ranked list.

Similarly, in the co-author dataset (Figure 4(b)), for 39% of advisees (824 out of 2099), the advisor is ranked among the highest authors related to that advisee. In addition to that, for about 75% (1582 out of 2099) of advisees, the advisor appears in the top 0.0008% (top 8 authors) in the advisee's ranked list. This percentage rises to 87% (1826 out of 2099) when we look at the top 0.0018% (top 20 authors).

[2]http://research.cs.wisc.edu/htcondor/

## 6.5 Time-Function (Time-F) Results

As described earlier, for each actor $x$, we rank the other actors by the time-scores calculated using Formula (2). Figure 4(a) reveals the results obtained by Time-F in detecting manager-subordinate ties in the Enron dataset. About 39% of manager-subordinate pairs can be detected precisely, that is, where the manager appears first within the subordinate's ranked list. This shows an improvement over RPR which detected only 32% correctly. In general, Time-F performs better than RPR in detecting manager-subordinate ties when we look at the top $x$% of the subordinate's ranked list where $x \leq 5.1$, i.e., the top 8 or fewer employees.

In the co-author dataset (Figure 4(b)), Time-F is better than RPR in detecting advisor-advisee relationships with a significant improvement of 20-30% in most cases. In 62% of cases, advisors are ranked first in the list of their advisees. This compares to only 39% in the case of RPR.

## 6.6 Filter-and-refine (FiRe) Results

In order to find the best cut-off position $k$, we tested all possible values of $k$ from 2 to 20. Tables 2 and 3 list the most interesting FiRe results using different cut-off values $k$ on the Enron and co-author datasets respectively.

In the Enron dataset, about 48% of manager-subordinate pairs can be detected with cut-off $k = 3$ or $k = 4$. This percentage decreases slightly to 45% when $k = 7$ but with better results for those relations where managers appear in the top 2 or 3 of their subordinate's ranked lists. Figure 4(a) compares between RPR, Time-F and FiRe. Clearly, FiRe is the best approach with significant improvement in detecting managers who are ranked in the top three of their subordinates' lists. For example, in 48% of manager-subordinate relations, managers come first in the ranked lists, compared to 39% and 32% detected by Time-F and RPR respectively.

In the co-author dataset, the best results are for a cut-off position of $k = 2$, with about 60% of advisor-advisee relationships being detected at rank one. However the cut-off at $k = 3$ is slightly better at retrieving advisors in the top three of their advisees' lists. It should be noted that by the definition of $LT_x(k)$ in Equation (5), the cut-off at $k$ may return a list of more than $k$ actors. This occurs when multiple actors have the same $TS$ scores around the cut off position, and explains the increase in detected relations for $k = 2$ when we look at the top two (81.35%) compared to the top three authors (82.97%). Comparing the three approaches in Figure 4(b), FiRe and Time-F perform similarly in detecting relations where the advisors come first in the ranked lists. They achieve 60% and 62% respectively. Time-F is preferable to RPR and FiRe in cases where advisors appear in the top three or more authors within the ranked lists.

In summary, considering the time-dimension of interactions between actors, as in Time-F, improves the results of detecting hierarchical relationships compared to structure-based approaches like RPR. Moreover, in some applications like email networks, hybrid models (FiRe) based on both network structure and the temporal patterns of interactions give further improvement over the pure time-based (Time-F) or structure-based models (RPR).

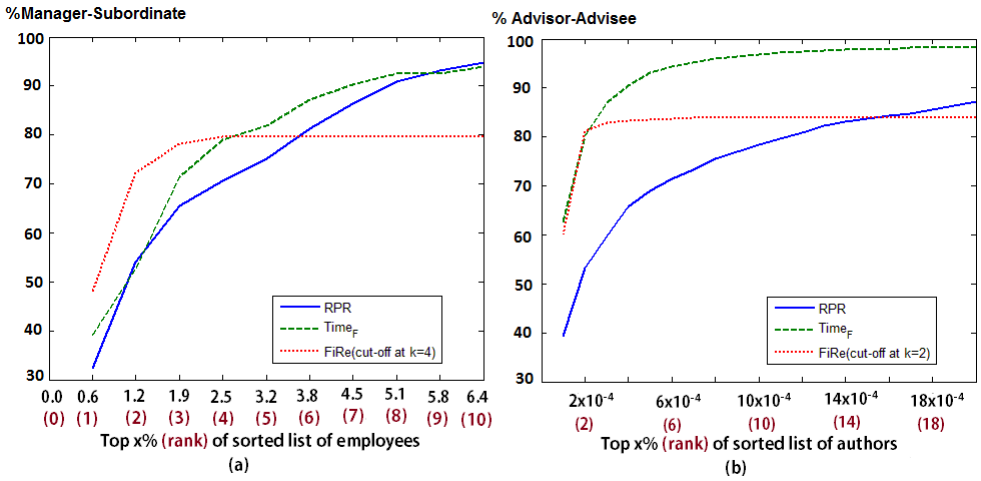


**Figure 4: Results of RPR, Time-F and FiRe on (a) the Enron dataset and (b) the co-author dataset.**

| cut off | $Pos(LTR_x, y)$ | | |
|---|---|---|---|
| | $= 1$ | $\leq 2$ | $\leq 3$ |
| 2 | 41.35 | 53.38 | 53.38 |
| 3 | **48.12** | 66.16 | 72.18 |
| 4 | **48.12** | **72.18** | 78.19 |
| 5 | 45.11 | 69.92 | 79.69 |
| 7 | 45.86 | 69.92 | **81.95** |
| 10 | 42.10 | 65.41 | 76.69 |

**Table 2: FiRe results for Enron dataset using different cut-off $k$.**

| cut off | $Pos(LTR_x, y)$ | | |
|---|---|---|---|
| | $= 1$ | $\leq 2$ | $\leq 3$ |
| 2 | **60.03** | **81.35** | 82.97 |
| 3 | 54.60 | 75.44 | **84.97** |
| 4 | 50.50 | 71.24 | 80.44 |
| 5 | 48.30 | 68.24 | 77.34 |
| 10 | 42.53 | 59.13 | 67.71 |
| 15 | 41.15 | 56.17 | 64.13 |

**Table 3: FiRe results for co-author dataset using different cut-off $k$.**

## 7. CONCLUSION

In this paper, we presented our work on inferring implicit off-line hierarchical ties between actors from their on-line interaction networks. We considered as examples detecting manager-subordinate relations between employees from their exchanged emails and discovering supervision relationships in academia by analysing the network of co-authored papers. Our experiments showed that studying the temporal patterns of interactions results in considerably better predictions of hierarchical relationships than models based on studying network structure alone (like Rooted-PageRank). In some applications, like email networks, heterogeneous models based on temporal and structural features of interaction perform even better than homogeneous models. In the future, we intend to develop a model that considers the time-dimension of interactions more precisely to improve the inference of hierarchical relationships between actors.

## 8. REFERENCES


[1] L. Backstrom and J. M. Kleinberg. Romantic partnerships and the dispersion of social ties: A network analysis of relationship status on Facebook. *CoRR*, abs/1310.6753, 2013.

[2] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30:107–117, 1998.

[3] M. Burke, L. A. Adamic, and K. Marciniak. Families on Facebook. In *ICWSM*, 2013.

[4] M. Gupte, P. Shankar, J. Li, S. Muthukrishnan, and L. Iftode. Finding hierarchy in directed online social networks. In *WWW*, pages 557–566. ACM, 2011.

[5] Z. Huang and D. K. J. Lin. The time-series link prediction problem with applications in communication surveillance. *INFORMS Journal on Computing*, 21(2):286–303, 2009.

[6] C. H. Hubbell. An input-output approach to clique identification. *Sociometry*, 28(4):377–399, 1965.

[7] Z. Katona, P. P. Zubcsek, and M. Sarvary. Network effects and personal influences: the diffusion of an online social network. *J. Marketing Research*, 48, 2011.

[8] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.

[9] P. Lazarsfeld, B. Berelson, and H. Gaudet. *The People's Choice: How the Voter Makes Up His Mind in a Presidential Campaign*. Columbia University Press, 1948.

[10] X. Li, B. Liu, and P. S. Yu. Time sensitive ranking with application to publication search. In *ICDM*, pages 893–898, 2008.

[11] J. Liebowitz. Linking social network analysis with the analytic hierarchy process for knowledge mapping in organizations. *J. Knowledge Management*, 9(1):76–86, 2005.

[12] M. E. J. Newman. Scientific collaboration networks. II. shortest paths, weighted networks, and centrality. *Physical Review E*, 64, 2001.

[13] M. E. J. Newman. The mathematics of networks. *The New Palgrave Encyclopedia of Economics*, 2007.

[14] J. Tang, T. Lou, and J. Kleinberg. Inferring social ties across heterogenous networks. In *WSDM*, pages 743–752, 2012.

[15] Z. Xiong, W. Jiang, and G. Wang. Evaluating user community influence in online social networks. In *IEEE TrustCom*, pages 640–647, 2012.

[16] V. Zlatic, G. Bianconi, A. Díaz-Guilera, et al. On the rich-club effect in dense and weighted networks. *The European Physical Journal B - Condensed Matter and Complex Systems*, 67(3):271–275, 2009.