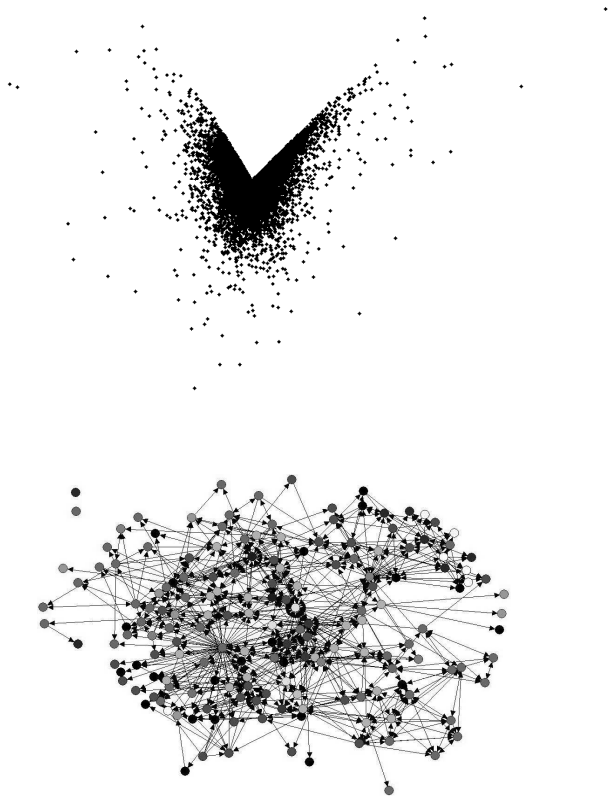
Workshop on Link Analysis, Counterterrorism and Security

at the SIAM International Conference on Data Mining

Sutton Place Hotel

Newport Beach, California, USA

23rd April, 2005



Workshop on Link Analysis, Counterterrorism and Security

at the SIAM International Conference on Data Mining
Sutton Place Hotel
Newport Beach, California, USA
23rd April, 2005

Organizers:
D.B. Skillicorn
K. Carley

Table of Contents

This workshop is the third on this topic at the SIAM International Conference on Data Mining. Research in areas such as link analysis, social network analysis, dynamic network analysis, and text analysis has a long history of use to understand how information flows in organizations, how people form relationships and connections, and how this affects decision making. These techniques have also been applied to understanding pathologies in organizations: how collusion and fraud reveal themselves in the links within and among organizations.

The growth of the Salafist terrorist movement, and in particular the attacks of September 11th, 2001, have moved research in this area from academia to an important part of many countries' defensive technology. Such techniques are of value in identifying key actors, locating experts, identifying unique patterns of transactions, characterizing the shape of and differences in terrorist groups, and locating areas of expertise.

It has been clear from the start that successfully discovering terrorism, fraud, or other covert activities requires analyzing large, complex, and messy datasets. Furthermore, the patterns in these datasets are usually small in scale and hard to pick out against the background of normal every day behavior. As Ted Senator has said, the problem is like trying to find a needle in a haystack of needle pieces. This creates difficult new problems for analysis techniques: pragmatic problems caused by the sheer size and complexity of the data, and discrimination problems, determining when some small variation in the structure of the data is potentially interesting. The situation is further complicated by the fact that such data is inherently messy reflecting the vast array of original data sources (e.g., news plus web plus email), biases in data collection, and intentional ambiguities (such as false identities).

There are two broad kinds of analysis. The first looks at the properties of individual objects, perhaps people or messages or journeys, and tries to detect those that are anomalous in some useful way. The second looks at the relationships between objects, and tries to find patterns in their connections that are anomalous. Again, there are two broad kinds of approaches. The first focuses on streams of data and tries to locate anomalies as new data arrives. The second focuses on the data as though it were a single block in time, a snapshot of the world, and tries to locate anomalies within this snapshot. The research reported here, is split between the two types of analysis but is more focused on the data as a snapshot rather than as a stream.

One of the problems for academic researchers has been the availability of appropriate datasets against which to try techniques. One such dataset is the online movie database; but this is a stylized archive of transactions and does not reflect the vagaries of everyday communication. In the wake of the collapse of Enron, a large set of email records was released by the U.S. Department of Justice. This created an opportunity for researchers to try their techniques in a realistic way on a database of actual everyday transactions and to compare the results. This workshop is one opportunity to do so.

First, it is clear the email is not quite like either spoken or formal written communication. Email tends to occupy a middle ground: less formal than other forms of writing, but more formal than speech. The Enron emails provide a chance to investigate, empirically, what the language of email is like. Second, emails have a sender and one of more receivers, and so represent a form of connection between people. It is natural to build various forms of graphs to capture these connections, and then to see what they can tell us about how communication works, and how it connects to relationships and to power. This is complicated by the fact that: a) the senders/receivers may have multiple identities (email ids) and b) the receivers may be groups such as mailing lists. Further, each sender and receiver can be further characterized by the domain from which they are sending and sometimes

by the role they play within a company (such as president or CEO).

Third, emails are written for a purpose - they are about something. Examining the content of real emails can tell us how information flows in an organization, how information reflects relationships, and also how word usage and style might reflect relationships and power.

Fourth, emails are timestamped, so it is possible to look at how email usage changes with time. This is particularly interesting because Enron was undergoing a change in leadership and the fraudulent scheme was unravelling during the period of time over which these emails were captured, and connections can perhaps be made between patterns in the emails, and activities in the outside world.

The results presented here are all preliminary. The sheer size and complexity of the dataset resulted in massive amounts of time being spent simply cleaning the data; e.g., eliminating copies of messages, identifying when the same person had multiple ids, and so on. Researchers engaged in looking at the data were often forced to rebuild or extend their software to account for the scale of data or for features in the dataset that were unanticipated (such as multiple email ids for the same person). As a result, the research reported on is as much about technology as it is about Enron, perhaps more so. Despite the preliminary level of the included research, much of interest has been observed. This is due to both the importance of the data and the value added by the new methodologies.

The papers in this session represent important advances in data-mining that, in many cases, merge machine learning techniques, link analysis, and social network analysis into new capabilities. To be sure, reading these papers provide some understanding of the massive changes Enron was undergoing. However, the insights here are nowhere as striking theoretically as the increase in capability afforded by the new methodologies. That being said, we note that the methodologies still need to be improved to be faster and more robust in the face of messy data. Outstanding issues remain such as those surrounding automated identification of aliases, experts, areas of discussion, automated ontology creation, and automated monitoring of streaming data.

We anticipate this email dataset will continue to be studied for many years. It represents a unique point in American history and an unprecedented level of access to daily information. We expect that future work will move to using a unified cleaned dataset. We also expect that future work will progress to advance not just the methodologies but our understanding of information flow, corporate planning and corporate decision making. Thus, while the current workshop is very methodologically focused, we anticipate that future ones will be as or more theoretically focused.

We would like to thank the members of the Program Committee for their support in publicizing the workshop, and for helping us to review the submissions. We would also like to thank others who took the time to review submissions, necessarily with short timelines. And, of course, we would also like to thank the researchers who submitted papers. We received many submissions and, given the limited time available for the workshop, had to reject many worthy submissions.

David Skillicorn
Kathleen Carley

Exploration of Communication Networks from the Enron Email Corpus[1]

Jana Diesner (diesner@cs.cmu.edu)
Kathleen M. Carley (kathleen.carley@cmu.edu)
Carnegie Mellon University

Abstract

The Enron email corpus is appealing to researchers because it is a) a large scale email collection from b) a real organization c) over a period of 3.5 years. In this paper we contribute to the initial investigation of the Enron email dataset from a social network analytic perspective. We report on how we enhanced and refined the Enron corpus with respect to relational data and how we extracted communication networks from it. We apply various network analytic techniques in order to explore structural properties of the networks in Enron and to identify key players across time. Our initial results indicate that during the Enron crisis the network had been denser, more centralized and more connected than during normal times. Our data also suggests that during the crisis the communication among Enron's employees had been more diverse with respect to people's formal positions, and that top executives had formed a tight clique with mutual support and highly brokered interactions with the rest of organization. The insights gained with the analyses we perform and propose are of potential further benefit for modeling the development of crisis scenarios in organizations and the investigation of indicators of failure.

Key Words: Enron, social network analysis, dynamic social networks, communication networks, DyNetML, ORA

1 Introduction

The Enron email corpus is appealing to researchers because it is a) a large scale email collection from b) a real organization c) over a period of 3.5 years. For research related to Social Networks, Organizational Theory, and Organizational Behavior this dataset is of particular interest and potential value because it enables the long term examination of interactions and processes within and among the entities of an organization. The Enron corpus contains a large amount of information on interaction, communication, knowledge, cognition, resources, tasks and relationships on an individual and group level in Enron. In order to explore and understand how these factors might have impacted the network, its design, culture, and life cycle, we need to extract and analyze this information in an effective and efficient way.

There is a growing body of research on various aspects of the Enron email corpus. To date, most publications have focused on Natural Language Processing (NLP) of the data: Klimt and Yang [17][18] and Bekkerman [2] explored the classification of emails, such as the organization of messages in user-defined folders and thread detection. Corrada-Emmanuel used the MD5 digest to generate mappings of the dataset, such as mapping of authors and recipients [8]. Shetty and Adibi [33] provide information on quantitative features of the corpus, such as the distribution of the number of emails per user and over time (months, years). They generated a social network that represents 151 Enron employees. In this network each exchange of at least 5 emails between any pair of agents across the entire time range (1998 to 2002) was considered as a link.

Essentially, the research community is exploring the Enron dataset from a mainly NLP perspective. In this paper we contribute to this initial investigation from a network analytic perspective: We describe how we enhanced and refined the Enron email database with respect to relational data. Moreover, we report on how we extracted network data from our instance of the corpus and demonstrate the application of various social network analytic techniques to the exploration of structural and behavioral features of the organization under investigation. The network analytic perspective enables us to investigate vulnerabilities of the system and its adaptivity to changing situations. The insights gained with the analyses we perform and propose are of potential further benefit for modeling the development of crisis scenarios in organizations and the investigation of indicators of failure. Note that the work presented in this paper is research in progress; the results of our sample study cannot be generalized for the Enron corporation or other organizations, but show what knowledge we can gain from analyzing an email corpus from a network analytic perspective and what kind of questions we can answer.

[1] This paper is part of the Dynamics Networks project in CASOS (Center for Computational Analysis of Social and Organizational Systems, http://www.casos.cs.cmu.edu) at Carnegie Mellon University. This work was supported in part by the Office of Naval Research (ONR), United States Navy Grant No. 9620.1.1140071 on Dynamic Network Analysis under the direction of Rebecca Goolsby. Additional support on measures was provided by the DOD and the NSF. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the National Science Foundation or the U.S. government. We thank Corinne Coen (SUNY, Buffalo) for her advice on this project, Eduard Hovy (USC, ISI) for pointing us to ISI's work on Enron, and the CASOS lab for their help on this work; especially Andrew Dougherty and Dan Woods.

Section 2 provides a synopsis of the Enron case and develops our research questions. Section 3 describes the dataset. In section 4 we report on how we refined the database and extracted relational data from it. Next we describe our methodology for analyzing the extracted data. Section 6 presents initial analyses results. Section 7 reports on the limitations and of our study. Section 8 points out directions for future work.

2 The Enron Case

Enron - What happened?

Enron was formed in 1985 under the direction of Kenneth Lay through the merger of Houston Natural Gas, a utility company, and Internorth of Omaha, a gas pipeline company. The company was based in Houston, Texas. Within 15 years Enron became the nation's seventh-biggest company in revenue by buying electricity from generators and selling it to consumers. The company quickly adapted to the deregulation of the energy market by positioning themselves as an energy broker: Enron identified areas where energy needs where higher than energy capacities, built power plants in such regions, sold the plants before their value diminished, and moved on to new areas with mismatches of power needs and capacities [28]. Later the company applied and expanded their middlemen skills and derivate trades to newer markets such as TV ad time and bandwidth. In 2002, Enron employed 21,000 people in more than 40 countries [10].

From 1985 on, Arthur Andersen, LLP (Andersen) had been Enron's auditor. Andersen earned tens of millions of dollars from accounting and internal and external consulting services for Enron, which was one of Andersen's largest clients worldwide. Enron employed many former Andersen workers.

In 1999, Enron officials began to separate losses from equity and derivate trades into "special purpose entities" (SPE); partnerships that were excluded from the company's net income reports. An example of such an SPE was Raptor, a liaison of Enron executives, who bought equity shares in two companies, New Power Co. and Avici, with loaned stock money from Enron. Enron profited from the increase of the value of the SPE's shares but had Raptor booking the losses, thus excluding them from their financial reports. The systematic omission of negative balance sheets and income statements from SPEs in Enron's reports resulted in an off-balance-sheet-financing system [28].

In December of 2000, president and chief operating officer Jeffrey Skilling took over the position of chief executive from Kenneth Lay. Lay remained chairman while the Enron stock hit a 52-week high of $84.87. In August 2001 Skilling surprisingly resigned, stating personal reasons for quitting. Lay was named as Enron's chief officer and CEO again in 2001 [20]. In the same month Sherron Watkins, Enron's Vice-President of Corporate Development who became famous as Enron's whistle-blower, wrote an anonymous letter to Lay in which she accused Enron of possible fraud and improprieties such as the SPEs [31]. Andersen knew of the information provided by Sherron Watkins.

In October 2001 the losses transferred from Enron to the SPE's totaled over $618 million and Enron publicly reported this amount as net loss for the third quarter. By the end of the year Enron disclosed a reduction of $1.2 billion in the value of shareholders' stake in the company. One of the people associated with the crash was Andrew Fastow, chief financial officer, who had supported Enron in inflating profits and hiding debts [28].

On October 31, 2001, the Securities and Exchange Commission (SEC) started an inquiry into Enron. Enron subsequently ousted Fastow and announced that the SEC investigation revealed that the amount of losses for the previous five years was actually $586 million. The market reacted with a fast and sharp drop of the value of Enron's shares to levels below $1 in November 2001. Being forced to transfer stocks in order to satisfy the losses, Enron became insolvent and filed for bankruptcy in December 2001. The fallout and investigations into the Enron collapse continued throughout 2002. Lay resigned as chairman and CEO in January of 2002, and less than two weeks later from the board [1].

Long before Enron's official insolvency, Andersen had possessed knowledge of Enron's organizational situation and financial performance but did not communicate the information to the public [28]. Andersen and Enron intentionally categorized hundreds of millions of dollars of shareholders equity that were a decrease as an increase. Andersen, who did some of Enron's internal bookkeeping, advised Enron not to refer to charges against the third quarter income of 2001 as non-recurring, but did not make this information available for the public. In 2000 Andersen's internal Senior Management already had rated Enron lower then they evaluated the client publicly. Before Enron released its notice of net loss, Andersen retained a New York based law firm from handling further Enron-related issues and took over all legal matters regarding Enron. In late October 2002, Andersen instructed Enron to destroy documentation related to Enron.

Andersen was indicted for altering, destroying and concealing Enron-related material and persuading others to do the same in March 2002 [36], convicted of obstruction in June 2002, and received a probationary sentence and a fine of

$500,000 in October 2002. In 2002 Andersen got banned from auditing public companies.

Lay, Fastow and former top aid Michael Kopper appeared before Congress in February of 2002; all three of them invoked the Fifth Amendment [10]. Skilling testified twice before Congress the same month, stating that he was unaware of any accounting problems. Fastow was indicted in October 2002. Ben Glisan Jr., a former Enron treasurer, pleaded guilty to conspiracy in September 2003, and became the first former Enron executive being imprisoned [1]. Fastow pleaded guilty in January 2004 [10]. His wife, Lea Fastow, and seven former Enron executives also got charged. In February 2004 Skilling got charged with fraud, conspiracy, filing false statements to auditors and insider trading [20]. In July of 2004 Lay surrendered to the FBI and was accused of participating in a conspiracy to manipulate Enron's quarterly financial results, making false and misleading public statements about Enron's financial situation, omitting facts necessary to make financial statements accurate and fair, civil fraud, and insider trading.

In March of 2003 Enron announced a plan to emerge from bankruptcy as two separate companies. In July the company filed a reorganization plan stating that most creditors would receive about one-fifth of the $67 billion they were owed.

Research on the Enron Case

Much information is available on the Enron case[2], including some details on organizational aspects of the company that might relate to its failure, such as a certain organizational culture. However, no studies of the case have been published yet in the Organizational Science and Social Networks literature.

The Board Investigation Committee stated in February 2002 that Enron's board may have been withholding critical information and had been unable to or prevented from providing checks and balances that would have been necessary to assure ethical business practices[26]. The Congressional Commission reported that Enron's culture encouraged employees to push the limits [26].

The Management Institute of Paris (MIP) identified Enron's and Andersen's senior managers as those in charge of Enron's failure. According to them, Enron's management misled the public, lacked moral leadership and ethics, and created an organizational culture of greed, secrecy and winner-take-all mentality. In 2001 Andersen evaluated Enron's financial statements as adequate and reliable and their financial conditions as fair [22].

Based on an article in Fortunes Magazine that explains the bankruptcy of over 257 companies in 2001 with managerial errors rather than with extra-organizational factors, which are usually claimed by the management, MIP points out ten executive errors that lead to Enron's failure [23]. These factors can be grouped into three categories: misperception of reality, risk-taking organizational culture, and improper crisis management.

Misperception of reality occurred in Enron on managerial level, because a) executives ignored bad news since it did not fit into their mental models of success that they had build up previously, b) mangers blinded out perceived problems instead of tackling them, and c) employees mitigated problems they reported to their supervisors for fear of the rogue character of Enron's managers (for example, Sherron Watkins having sent her letter anonymously to Lay). Instances of Enron's risk-taking culture are the foundation of SPE's, the overdosing of risk by not providing liability for the SPE's losses, and the greedy profit taking without disclosure. Enron's improper crises involved the implementation of ad-hoc strategies, hoping for a quick solution of all difficulties and lacking a thorough analysis of the problem.

While first thoughts about the relationship between Enron's risk-pushing organizational culture in connection with managerial errors and the company's failure are being released, no network analytic studies have been published that explore the social network phenomena in Enron (with exception for the social network generated by Shetty and Adibi [33]) .

Network analysis focuses on the relations among and between entities in a social or organizational system (see for example [29][38]). In our case, the system is Enron and the entities are former Enron employees. In a social network the entities are represented as nodes, and the relations between them as edges or links. We base the research presented in this paper on the assumption that the relations among Enron's employees are represented in the exchange and content of the emails that are contained in the Enron corpus. In our study we focus on the analysis of the exchange of emails. We refer to this type of networks as communication networks because these networks represent flow of messages among communicators across space and time [24]. Since the messages are sent from one agent to one or multiple other agents, the resulting networks are directional or digraphs.

[2] See material from agencies such as SEC [30], Federal Energy Regulation Commission (FERC) [12], United States Department of Justice (DOJ), Commodities Futures Trading Commission (CFTC) [6], General Accounting Office (GAO), Investigative Committee of the Board of Directors of Enron [26], and management related organizations [15].

The lack of research on Enron from a network analytic perspective motivates our research questions:

What are the structure and properties of the communication networks in Enron? How do these features relate to other networks?

Who are key players or critical individuals in the system? (On the concept of key players see [3]).

How do structure and key players change over time?

Our research questions are of an explorative nature and aim to gain a first understanding of relations between individuals in Enron. Answers to these questions will provide researchers with knowledge that can help to understand and explain this particular organization and relate this information to Enron's life cycle of success, crisis and bankruptcy. The network analytic perspective enables the investigation of vulnerabilities of the system and its adaptive capabilities to changing situations. Furthermore, the relational data that we extract and its analysis could be deployed to further develop theories or validate hypotheses about the evolution of communication networks.

3 Data

There is not *the* Enron email corpus available, but multiple instances of it. The Federal Energy Regulatory Commission (FERC) originally posted the Enron email database on the internet in May of 2002 to enable the public to understand why FERC investigates Enron [12]. The database consists of 92% of Enron's staff emails. FERC collected a total of 619,449 emails from 158 Enron employees, mainly from senior managers. Each email contains the email address of the sender and receiver, date, time, subject, body and text. Attachments were not made available. FERC's version of the database had a lot of integrity problems. Leslie Kaelbing from MIT then purchased the dataset. Later a group of people at SRI, notably Melinda Gervasio, collected and prepared the data for the CALO project [34]. The SRI group corrected most of the integrity problem and made the dataset available.

William Cohen from CMU put the dataset online for researchers in March 2004 [7]. This version of the database contains 517,431 distinct emails from 151 users. The emails are organized in 150 user folders that have further subfolders; with the total number of folders in the corpus totalling 4700. The corpus has a size of 400Mb. Some messages were deleted "as part of a redaction effort due to requests from affected employees" [7]. Invalid email addresses were converted to addresses of the form user@enron.com when a recipient was specified and to no_address@enron.com when no recipient was specified.

Andres Corrada-Emmanuel from the University of Massachusetts further explored the dataset by using the MD5 digest of the body of the emails. He found out that the corpus actually contains 250,484 unique emails from 149 people [8].

The version of the dataset that we are using was provided by Jitesh Shetty and Jafar Adibi from ISI [33]. The ISI people cleaned up the dataset by dropping emails that were blank, duplicates of unique emails, had junk data, or were returned by the system due to transaction failures. The resulting corpus contains 252,759 emails in 3000 user defined folders from 151 people. Shetty and Abidi put the information in a MySql database that contains four tables, one for each of the entities of employees, messages, recipients and reference information. We chose this version of the corpus for our work, because the process of cleaning the dataset seems very helpful to us and is well documented. Furthermore, the structure and content of the MySQL database met our needs.

The database contains many emails by individuals who were not involved in any of the actions that are subject of the Enron investigation.

4 Database Refinement and Extraction of Relational Data

In order to perform network analysis on the Enron corpus, it is necessary to extract relational data. The relations among and between the entities in Enron are reflected in a) the email exchanged between the employees (communication networks) and b) the actual content of those messages. In this paper we concentrate on the extraction and explorative analysis of the first type of data. All database work and data extraction was performed on a Linux machine with Perl modules that we wrote for this purpose.

The data in the corpus is multi-mode (e.g. work relationship, friendship), multi-link (connections across various meta-matrix entities) and multi-time period. Nodes and edges can have multiple attributes such as the position and location of an employee or the types of relationships between two communication partners (multi-mode). We refer to data that is multi-mode, multi-link and multi-time period in which both nodes and edges can have attributes that carry information on how to interpret, evolve, and impact these nodes and edges as "rich" data. In order to adequately represent and analyze the information contained in the corpus we need a data format that can handle rich social network data and can be used as input and output of multiple analysis tools that we consider to use. We chose to use DyNetML as the data format because it meets our data format requirements [35]. DyNetML is an XML based interchange language for relational data. A

DyNetML file can represent an arbitrary number of node sets and graphs. Node sets group together nodes of the same type, e.g. agents, complete with any rich data such as an agent's position or location. Each graph consists of a set of edges that connect nodes, complete with any rich data attached to the graph itself or any of its edges.

Database Refinement

DyNetML files for the representation of communication networks require data from three tables in the ISI database: The message ID, which includes time information, the sender, and the recipient. The information provided on the individuals is their first and last names and one email address. More information on properties of the individuals would enable a more thorough analysis and deeper understanding of processes in Enron. Such properties can be represented as attributes of nodes that represent agents in DyNetML. We found three additional sources of information on some of the Enron employees: A file with the positions of former employees from ISI (ISI position file) [32], a list with job information from FERC/ Aspen (FERC position file) [11], and a list from FERC/ Apsen with information on people's location (FERC location file) [13]. Note, most of the information on FERC's Western Energy Markets investigation is hosted on Aspen Corporation websites.

The ISI position file lists the names of 161 Enron employees, and for 132 of them it provides position information. ISI gathered this information from various sources, mostly from Federal Court documents which were publicly released. For 29 people no status information is provided because they, according to Shetty, were not involved in the Enron case and did not hold high posts in the company, or were employed for a only short period of time. In the social network generated by Shetty and Adibi those 29 people are assigned to the position of an employee (Table 1)[3]. The FERC position file is a list of authorized traders that contains names, positions, a few locations and trade related information on individuals from Enron and probably other companies. The FERC location file is an interoffice memorandum sent by John Lavorato to Donna Lowry from Risk Assesment and Control on October 12, 2001. In this file, people are sorted by locations – East, Central, Texas, West and Canada.

[3] The ISI position file contains two sets of names that seem semantically highly similarity: Micheal Swerzzbin/ Vice President; Mike Swerzbin/ Trader; James Schweiger/ Vice President; Jim Schwieger/ Trader. We were skeptical if Swerzzbin/ Swerzbin and Schweiger/ Schwieger were distinct individuals, therefore we matched those names against both FERC files. Based on this comparison we selected Mike Swerzbin and Jim Schwieger as unique individuals, because they appeared in the FERC files, and dropped Micheal Swerzzbin and James Schweiger, because they were not listed in the FERC files.

We added the position and location information to a new instance of the Enron database that we built. We refer to our instance of the database as the Enron CASOS database. We realized that in many cases the spelling of names did not match between the files from ISI, FERC and the database. In order to find the names in the database that are most similar to the spellings in the ISI and FERC files we used a semantic similarity algorithm [36][21] implemented in the String Similarity Perl module [19], and ran it against the database. The similarity function computes a similarity value between 0 (no similarity) and 1 (identical strings), based on how many edits are necessary to convert one string into another. We output the 25 highest scoring suggestions from the module and picked the one that we manually evaluated to represent the same name that is provided in the database. After we had identified the matching pairs we added the position and/ or location information to these names as provided in the ISI and FERC files to the database while maintaining the spelling of the names as originally defined in the database. During this process we encountered various cases of conflicting information: In 36 cases we had different position information from ISI and FERC. We assume that this is because people got promoted or changed positions. Since we did not have time information for both of the files, our default was to pick the higher position. The location information in the two FERC files was conflicting in five cases. We picked the information from the location file because it had a date on it, which was in the middle of the crisis, and seemed more focused on location information. After using the heuristics described here we enhanced our instance of the database with the position and location information.

Overall, we identified 15 unique job titles that we associated with 212 employees (Table 1), 5 unique locations that relate to 67 people, 102 employees for which we have position and location data, and 227 employees for which we have either position or location information. For five of the 29 people that ISI had no position information on we were able to identify a job title. The further data adjustment and analyses in this paper mainly concentrate on the 227 employees whose names and/ or positions we know. A file with detailed information on this subset of people such as their first and last names, position, additional information on the position, location and source of this information is available from the authors but was not included in this paper to protect the individuals' privacy. We use this subset as a point of departure for our work on the Enron data, and

will include more people who appear in the database once we obtain more information on them.

Table 1: Number of Individuals per Position

Position	ISI position file	ISI social Network	CASOS Enron database
Analyst	0	0	10
Associate	0	0	5
CEO	4	4	4
Director	23	12	27
Employee	40	85	69
Head	0	0	2
In House Lawyer	3	3	3
Manag. Director	5	3	6
Manager	13	10	31
President	4	4	4
Specialist	0	0	9
Sr. Specialist	0	0	17
Trader	12	12	9
Treasury Support	0	0	2
Vice President	28	18	29
Total	**132**	**151**	**227**

Next we normalized the email addresses for the subset of 227 people. We assumed that people might have more than the one email address specified for them in the ISI corpus. Note that the spelling of emails in the database matches the spelling in the ISI position file. Corrada [8] provides a list of 31 email addresses that mainly resemble the addresses in the ISI file, but gives two addresses for only two out of 29 individuals. We further explored this issue by using the similarity function described above to search for all email addresses ending with @enron.com for addresses similar to those specified in the employeeList table in the ISI database. The module identified the 25 highest scoring hits per address, and we manually vetted them. We found that a similarity greater than 0.7 usually indicates a match and selected these by default prior to review. Table 2 provides quantitative information on the process of email normalization.

Table 2: Statistics of Email Address Normalization

	Emails referring to 227 agents	Emails added	Emails dropped
Sum	429	92	41
min	1	0	0
max	8	3	8
Average	1.89	0.41	0.18
STD	1.18	0.71	0.72

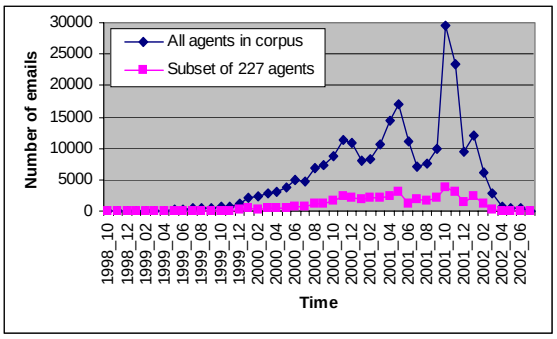
To summarize our work on the database, we have refined it by resolving name ambiguities and enhanced its information by adding the position and/ or location of 227 individuals, as well as normalizing their email addresses.

Extraction of Communication Networks

Next we extracted DyNetML files that represent the communication among the subset of 227 people. Out of the 227 individuals, a union of 209 people exchanged emails amongst each other. We time sliced our data in order to enable longitudinal analysis[4]. We decided to time slice the corpus on a monthly basis from October 1998 to July 2002, as this seemed to entail time spans in which major events occurred. This resulted in 46 DyNetML files that represent the agents as nodes and exchange of emails between them as edges. The number of agents in each file can differ since the size of the population can vary from month to month. Each edge denotes a directed relation of type agent to agent. The edges are weighted by the cumulative frequency of emails exchanged between individuals per month.

Figure 1 shows the total number of emails sent by all individuals in the corpus as well as by the people in our subset across months. Both curves show peaks in the amount of communication; some of them can be related to events in the organization. The highest peaks occurred in October 2001 (29,556), the month in which the Enron crisis broke out, November 2001 (23,441), when the investigations were under way, and May (16,986) and April (14,348) 2001. The low points, which are in January and February 2000 and from August to September, might be explained as being vacation periods. The curve for the subset resembles the pattern of the curve for the entire corpus.

Figure 1: Number of Emails Sent per Month



[4] The time slicing returned 327 emails from the entire corpus with invalid dates such 2044-01 or 0001-12. Since no correct date information was given in those emails we excluded those emails from further analysis. This reduced the corpus by 0.13% to 252,432 emails.

5 Methodology

We use ORA [5] to analyze the communication networks. Since we have position information on agents available we can compare the formal and informal organizational structure. We are also able to explore changes in the network over time by comparing a network from a month during the Enron crisis with a network from a month in which no major negative happenings are reported and where the organization seemed to be on a successful path. We picked October 2000 and 2001 for this comparison. We first run an intel report in ORA that computes network analytic measures on a graph level and identifies key agents in the network. Next we run an ORA context report that compares the graph level measures from the intel report for Enron with values for real networks stored in a CASOS database as well as with numbers computed on a directed uniform random graph of identical size and density as the Enron networks. Then we run an ORA risk report that identifies critical individuals who bear risks for an organization. The risk is computed for every agent as well as the entire network with respect to the agents' communication, performance, interaction, and redundancy. This report allows researchers to explore the distribution of a particular type of risk across an organization, thus identifying systemic versus individualistic problems.

6 Results

Figures 2 and 3 show the network structure by position for Oct. 2000 (160 agents) and 2001 (174 agents). The visualizations were generated with the NetDraw software [4]. Both graphs contain only a few isolates (one in Oct. 2000, 2 in Oct. 2001), which represent individuals who are not connected to others. ORA's intel report reveals that the Oct. 2001 graph is denser than the Oct. 2000 graph: The Oct. 2000 network has a lower overall completeness, expressed in the value of density (0.018), than the Oct. 2001 network (0.031). Mathematically densities range form 0 to 1, with higher values indicating denser graphs (for more details on network analytic measures see [5][38]). Looking at the number of weak or undirected components (2 in Oct. 2000, 3 in Oct. 2001) we learn that in both graphs all individuals, except for the isolates, are in one component. This means that in both networks each person can reach each any other person. Components are maximally connected subset of nodes, also referred to as subgraph. Weak components do not consider directionality of a link, whereas strong components take a link's directionality into account. The existence of components indicates that a graph is disconnected. The number of directed components is higher for Oct. 2000 (96) than for Oct. 2001 (39). This result suggests that during the crisis there are fewer disconnected subgroups of people who mutually exchange emails than in a normal month. The values of density and number of strong components indicate that during the crisis the communication among Enron employees has been intensified and spread out through the network in comparison to a month before the crisis.

Figure 2: Communication Network October 2000

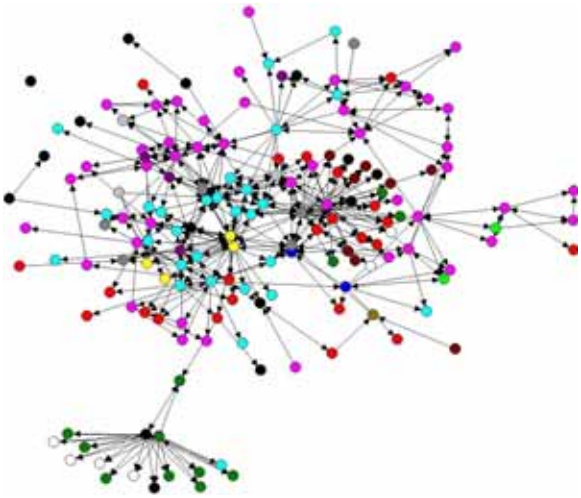
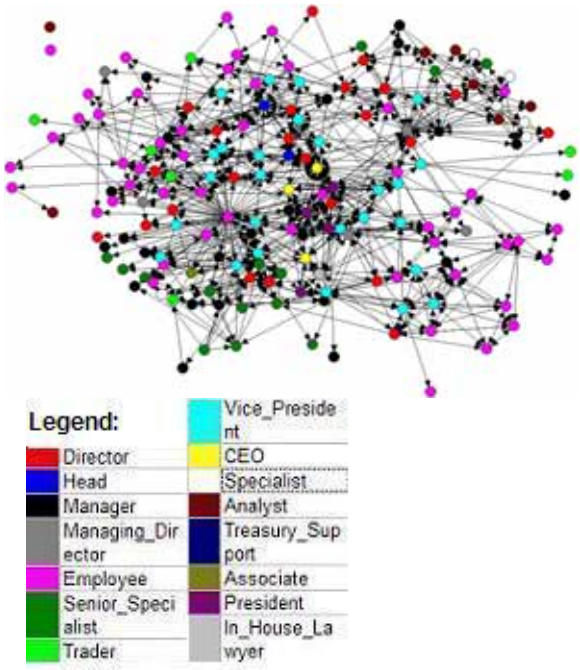


Figure 3: Communication Network October 2001



In order to put graph level measures for Enron into a broader context we run an ORA context report. Graph level centralization measures express the degree to which single actors have high

Table 3: Graph Level Measures in Comparison

Measure	Oct. 2000	Oct. 2001	Social Networks	Interpretation: On average ...
Betweenness Centrality	0.008	0.012	0.047	there are fewer paths by which information can flow from any one person to any other person in this group compared to other groups.
Closeness Centrality	0.031	0.253	0.380	it takes more steps for information to get from any person in this group to any other person in this group compared to other groups.
Eigenvector	0.046	0.055	0.165	this group is less cohesive than other groups.
Total Degree	0.018	0.031	0.284	each person in this group has fewer connections to others than people in other groups.
Strong Components	96	39	8.455	there are more components in this group than in other groups: i.e. it is more disconnected.

importance or prominence in a network and others have low centrality. Thus, graph centrality represents the heterogeneity or dispersion of the agents' centralities in a network[5]. The ORA results (Table 3) show that both Enron networks are less centralized than other networks, and that the Oct. 2000 graph is less centralized than the Oct. 2001 graph. These findings suggest that during the crisis the inequality of the importance of the employees, the amount of communication, and the group cohesion increased. The results also suggest that a highly segmented workforce with little cross communication may have been a factor that supported the frauds in Enron.

In order to identify the most important people in the network centralization measures can be computed on an individual level. Table 4 shows the 5 individuals who score highest in the Oct. 2000 and Oct. 2001 network with respect to the following centrality measures: Closeness centrality describes how close an actor is to all other actors. Betweenness centrality measures how often an actor is positioned on the shortest path between any other pair of actors. Eigenvector centrality tells us how close an actor is to other actors who are important with respect to degree centrality, and an actors' degree is the number of other actors directly linked to him or her. Since the Enron networks are directed, we split up centrality into outdegree (actors adjacent from an actor) and indegree (actors adjacent to an actor). Table 4 contains a union of 21 distinct people (13 distinct ones in Oct. 2000, 14 distinct ones in Oct. 2001), and 6 of them appear in both months. The intersection of individuals per measure in Oct. 2000 and Oct. 2001 is low and varies between 0 and 3. For the people who appear in both months their position in the ranking changes as often as it remains the same (4 times) from Oct. 2000 to Oct. 2001. Looking at the key players' formal positions the results show that for closeness centrality people with lower positions appear more often among the most central individuals in Oct. 2001 than in Oct. 2000. This observation does not apply to the other measures, but in general people with higher positions are more likely to be key players in this organization. Analyzing the values for closeness centrality for all 209 agents across all 46 months (Figure 4) reveals that the values per individuals are less different from each other than for other

Table 4: Key Players per Centrality Measures

October 2000			October 2001		
Value	**Name**	**Position**	**Value**	**Name**	**Position**
Closeness Centrality					
0.07	W. Stuart	Manager	0.21	S. Beck	Employee
0.07	D. Delainey	CEO	0.20	L. Kitchen	President
0.07	C. Dorland	Manager	0.19	S. Kean	VP
0.07	J. Derrick	Lawyer	0.19	S. White	Employee
0.07	T. Belden	Mang. Dir.	0.18	J. Dasovich	Employee
Betweenness Centrality					
0.11	D. Delainey	CEO	0.24	L. Kitchen	President
0.10	R. Sanders	VP	0.16	S. Beck	Employee
0.08	T. Belden	Mang. Dir.	0.13	T. Belden	Mang. Dir.
0.08	J. Lavorato	CEO	0.10	J. Lavorato	CEO
0.08	J. Dasovich	Employee	0.07	M. Grigsby	Head
Eigenvector Centrality					
0.60	J. Dasovich	Employee	0.69	J. Dasovich	Employee
0.54	J. Steffes	VP	0.52	J. Steffes	VP
0.41	M. Hain	Lawyer	0.40	R. Shapiro	VP
0.31	R. Shapiro	VP	0.23	S. Kean	VP
0.19	R. Sanders	VP	0.13	B. Tycholiz	VP
In Degree Centrality					
0.80	J. Steffes	VP	0.77	R. Shapiro	VP
0.46	R. Shapiro	VP	0.76	J. Lavorato	CEO
0.42	T. Belden	Mang. Dir.	0.66	B. Tycholiz	VP
0.36	M. Taylor	Employee	0.66	J. Steffes	VP
0.33	R. Sanders	VP	0.49	L. Kitchen	President
Out Degree Centrality					
1.08	J. Dasovich	Employee	1.63	D. Delainey	Employee
1.01	M. Hain	Lawyer	1.51	M. Grigsby	Head
0.96	T. Jones	Employee	1.04	B. Williams	Analyst
0.81	D. Delainey	CEO	0.90	S. Beck	Employee
0.48	T. Belden	Mang. Dir.	0.76	J. Steffes	VP

[5] On graph and node level, betweenness and closeness centrality vary between 0 and 1. Eigenvector and degree centrality can reach values higher than 1. The higher the value the more central is a network or an agent in a network.

measures (for example eigenvector centrality Figure 5). These results suggest that in 2000 Enron had a segmented culture with directives being sent from on-high and sporadic feedback. By 2001, the VP's and other executives had formed a tight knit clique supporting each other and whose interactions with the rest of Enron are highly brokered.

Table 5: Emails Exchanged per Month

	October 2000		October 2001	
Position	**sent**	**received**	**sent**	**received**
CEO	71%	29%	27%	73%
President	58%	42%	53%	47%
VP	38%	62%	44%	56%
Man. Dir.	43%	57%	57%	43%
Director	8%	92%	41%	59%
Head	57%	43%	79%	21%
Manager	53%	47%	42%	58%
Lawyer	72%	28%	52%	48%
Sr. Specialis	27%	73%	45%	55%
Specialist	0%	100%	29%	71%
Analyst	20%	80%	61%	39%
Associate	20%	80%	50%	50%
Employee	55%	45%	57%	43%
Trader	62%	38%	32%	68%

To further explore the relationship between positions and different situations in the company as well as the correspondence of the formal position network with the informal one we compared the amount of emails exchanged between positions (Tables 5, 6) for October 2000 and 2001.

Table 5 indicates that in contrast to Oct. 2000 in Oct. 2001 the CEOs, Heads, Managers and Traders sent more emails than they received, whereas the Managing Directors and Analysts received more messages than they sent. The major shift from 2000 to 2001 is that in 2000 higher rank positions tended to be directive (send more than receive) whereas by 2001 they became consumers (receive more than send). The major exception here are the VP's who have always been consumers and if anything became more directive.

The results in Table 6 show that high ranking positions (1 to 6 and 8 in Table 5) perform more top-down communication than the send information to higher ranks. In contrast, lower ranks send more communication up the hierarchy or within the same rank. Table 6 suggests that during the crisis 9 out of 12 positions communicate less with the same position or rank than they did in Oct. 2000. The differences of the percentages of emails sent to higher and lower ranks are less in Oct. 2001 than in Oct. 2000. Those findings indicate that during the crisis the communication has been more diverse with respect to formal positions than during a normal month. Furthermore, in contrast to Oct. 2000 in Oct. 2001 the Heads tended to communicate more often with lower ranks than with higher ranks and the Sr. Specialists more often sent messages to higher ranks than to lower ranks.

7 Limitations

The main limitation of our study is that we have not validated the relation data we have extracted and analyzed yet. In order to perform validation we will compare our data and findings against material from reliable sources such as reports and press articles on the Enron case, letters from and interviews with former Enron employees, and information from other people with direct insight into the company. Once we have such material we also will evaluate the extracted networks by analyzing what portion of the relevant links we have captured (recall) and what portion of the captured links is actually relevant (precision).

We note that the results presented herein cannot be generalized for the Enron organization or other corporations since we analyzed only two time points and a subset of 227 people.

8 Conclusion and Future Work

In this paper we have described how we enhanced and refined the Enron database. We have reported on the extraction of relational data from our instance of the database. Our initial results, which are based on snapshots of Enron's communication network at 2 time points, suggest that in Oct. 2001 the network had been denser, more centralized and more connected than in Oct. 2000. We also learned that about half of the people who were key players in Oct. 2000 were also the most important in the network of Oct. 2001. Our data suggests that during the crisis the communication among Enron's employees had been more diverse with respect to people's formal positions and that the top executives had formed a tight clique with mutual support and highly brokered interactions with the rest of organization.

In our future work we will consider all points of time that we extracted network data for and a larger set of people in order to learn more about this network and how its properties and entities relate to various phases of the company's life cycle of success, crisis and failure.

In the future we will analyze the actual content of the emails via Network Text Analysis [25][9] in order to explore the perception of the company's situation on an individual and group level, as well as across time. We will extract these perceptions as mental models, which are representations of the reality that people use to make sense of their surroundings [14][27].

Figure 4: Closeness Centrality of 209 Agents over Time

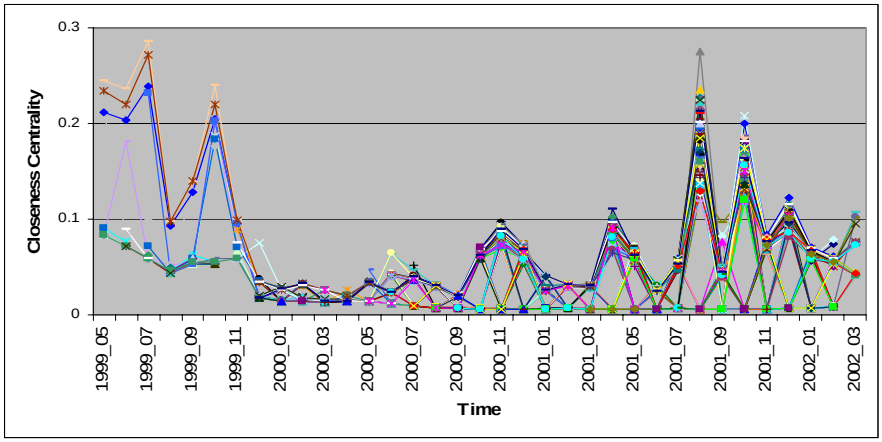


Figure 5: Eigenvector Centrality of 209 Agents over Time

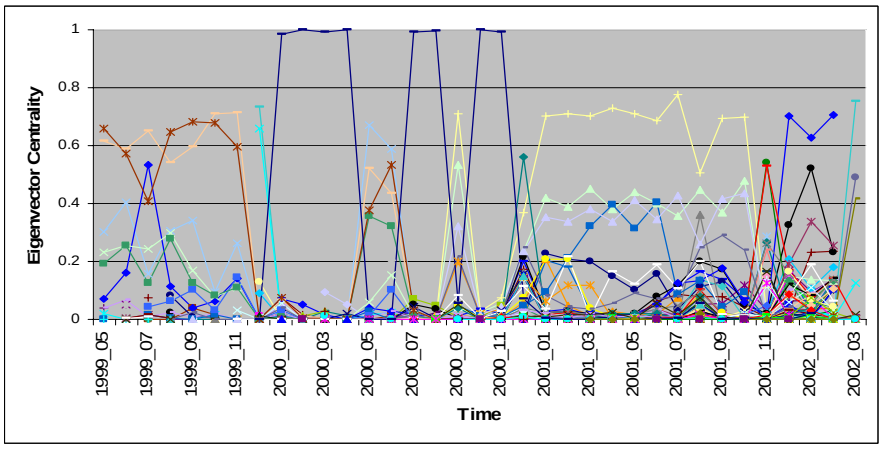


Table 6: Emails Sent to Positions

		October 2000			October 2001		
Rank	**Position**	**higher rank**	**lower rank**	**same pos. & same rank**	**higher rank**	**lower rank**	**same pos. & same rank**
1	CEO	NA	83%	17%	NA	100%	0%
2	President	12%	85%	3%	26%	54%	20%
3	VP	9%	58%	33%	14%	45%	41%
4	Man. Dir.	20%	75%	5%	30%	69%	1%
5	Director	27%	64%	9%	35%	43%	22%
6	Head	56%	31%	13%	43%	50%	7%
7	Sr. Specialist	6%	28%	66%	54%	30%	16%
8	Lawyer	89%	10%	1%	87%	13%	0%
9	Manager	18%	24%	59%	20%	49%	31%
10	Specialist	0%	0%	0%	17%	34%	49%
11	Analyst	40%	NA	60%	60%	NA	40%
11	Associate	100%	NA	0%	100%	NA	0%
11	Employee	40%	NA	60%	53%	NA	47%
11	Trader	NA	NA	98%	38%	NA	62%
11	Treas. Support	0%	0%	0%	100%	NA	0%

Mental models can be conceptualized as cognitive constructs that help researchers to gain an insight into how knowledge and information are represented in people's minds [16]. Since organizational culture is also represented in messages [24], we also will analyze the mental models to learn about Enron's culture.

References

[1] *A Chronology of Enron Corp.* (2004). NewsMax Wires. Retrieved October 13, 2004, from http://www.newsmax.com/archives/articles/2004/7/8/110332.shtml

[2] Bekkerman, R. (n.d.). Retrieved November 4, 2004, from http://www.cs.umass.edu/~ronb/

[3] Borgatti, S. P. (2004). The Key Player Problem. In R. Breiger, K. M. Carley, & P. Pattison (Eds.), *Dynamic Social Network Modeling and Analysis: 2002 Workshop Summary and Papers* (pp. 241-52). Washington, DC: National Academies Press.

[4] Borgatti, S.P. (2002). *NetDraw1.0. Graph Visualization Software*. Harvard: Analytic Technologies.

[5] Carley, K.M., & Reminga, J. (2004). *ORA: Organization Risk Analyzer*. Technical Report, Carnegie Mellon University, School of Computer Science, Institute for Software Research International, URL: http://www.casos.cs.cmu.edu/projects/ora/publications.html

[6] *CFTC Enron Information Link Page*. (2003). Retrieved October 9, 2004, from http://www.cftc.gov/enf/enron/enfenrondefault.htm

[7] Cohen, W.W. (n.d). *CALD, CMU*. Retrieved October 5, 2004, from http://www-2.cs.cmu.edu/~enron/

[8] Corrada-Emmanuel, A. (n.d.). *Enron Email Dataset Research*. Retrieved October 5, 2004, from http://ciir.cs.umass.edu/~corrada/enron/

[9] Diesner, J., & Carley, K.M. (2005). Revealing Social Structure from Texts: Meta-Matrix Text Analysis as a novel method for Network Text Analysis. In V.K. Narayanan & D.J. Armstrong (Eds.), *Causal Mapping for Information Systems and Technology Research: Approaches, Advances, and Illustrations*, (pp. 81-108). Harrisburg, PA: Idea Group Publishing.

[10] *Enron Scandal at a Glance.* (2002). BBC News. Retrieved October 13, 2004, from http://news.bbc.co.uk/1/hi/business/1780075.stm

[11] *FERC position file*. (n.d.). Retrieved October 10, 2004, from http://ferc.aspensys.com/FercData/Miscellaneous%20CD's/Box005/Response%20to%20Request%2015/RAC/Compliance/Authorized%20Trader%20Lists/Authorised%20Traders%20List5_11_01.pdf

[12] *FERC Western Energy Markets - Enron Investigation, PA02-2*. (n.d.). Retrieved October 18, 2004, from http://www.ferc.gov/industries/electric/indusact/wem/pa02-2/info-release.asp

[13] *Ferc/ Apsen Location file*. (n.d.). Retrieved November 4, 2004, from http://ferc.aspensys.com/FercData/Miscellaneous%20cd's/Box005/Response%20to%20Request%2015/RAC/Compliance/Authorized%20Trader/Authorized%20Trader%20Memos%20Dtd%2010-01/North%20American%20Natural%20Gas%2010-01.pdf

[14] Johnson-Laird, P. (1983). *Mental Models*. Cambridge, MA: Harvard University.

[15] Kefgen, K., & Kogen, M. (2002). *Enron Anyone?* HVS International. Retrieved November 4, 2004, from http://www.hvsinternational.com/emails/execsearch/hospitality/7-26.htm

[16] Klimoski, R., & Mohammed, S. (1994). Team mental model: Construct or metaphor? *Journal of Management* 20, 403-437.

[17] Klimt, B., & Yang, Y. (2004). Introducing the Enron Corpus. First Conference on Email and Anti-Spam (CEAS), Mountain View, CA. Retrieved October 14, 2004, from http://www.ceas.cc/papers-2004/168.pdf

[18] Klimt, B., & Yang, Y. (2004). The Enron Corpus: A New Dataset for Email Classification Research. *European Conference on Machine Learning*, Pisa, Italy.

[19] Lehmann, M. (n.d.). *String Similarity*. From http://search.cpan.org/~mlehmann/String-Similarity-1/Similarity.pm

[20] *Lights out at Enron.* (2003). CBSNews.com. Retrieved October 13, 2004, from http://www.cbsnews.com/stories/2003/02/06/60minutes/main539719.shtml

[21] Meyers, E.W. (1986). An O(ND) Difference Algorithm and its Variations. *Algorithmica*, 1(2).

[22] MIP. (2002). Enron: *Who is really to blame?* Retrieved November 11, 2004, from http://www.mip-paris.com/knowledge/article.asp?id=21.

[23] MIP. (2002). *Fortune Magazine's List of 10 Corporate Sins*. Retrieved November 11, 2004, from http://www.mip-paris.com/knowledge/article.asp?id=132

[24] Monge, P.R., & Contractor, N.S. (2003). *Theories of Communication Networks*. New York: Oxford University Press.

[25] Popping, R. (2000). *Computer-assisted Text Analysis*. Thousand Oaks, CA: Sage Publications.

[26] Powers, W.C. (2002). *Report of Investigation, By the Special Investigative Committee of the Board of Directors of Enron Corp*. Retrieved November 4, 2004, from http://news.findlaw.com/hdocs/docs/enron/sicreport/sicreport020102.pdf

[27] Rouse, W.B., & Morris, N.M. (1986). On looking into the black box; prospects and limits in the

search for mental models. *Psychological Bulletin* 100, 349-363.

[28] Sanborn, R. (n.d.). *Enron*. Retrieved November 4, 2004, from http://www.hoylecpa.com/cpe/lesson001/Lesson.htm

[29] Scott, J. (2000). *Social Network Analysis*. London: Sage, 2nd edition.

[30] *SEC Spotlight on Enron*. (n.d.). Retrieved November 4, 2004, from http://www.sec.gov/spotlight/enron.htm

[31] *Sherron Watkins eMail to Enron Chairman Kenneth Lay*. (2002). Retrieved November 11, 2004, from www.itmweb.com/f012002.htm

[32] Shetty, J., & Adibi, J. (n.d.). *Ex employee status report*. Retrieved November 4, 2004, from http://www.isi.edu/~adibi/Enron/Enron_Employee_Status.xls

[33] Shetty, J., & Adibi, J. (n.d.). *The Enron Dataset Database Schema and Brief Statistical Report*. Retrieved November 4, 2004, from http://www.isi.edu/~adibi/Enron/Enron_Dataset_Report.pdf

[34] *SRI International*, CALO (Cognitive Assistant that Learns and Organizes). (2004). Retrieved November 4, 2004, from http://www.ai.sri.com/project/CALO

[35] Tsvetovat, M., Reminga, J., & Carley, K.M. (2003). *DyNetML: Interchange Format for Rich Social Network Data*. CASOS Technical Report, Carnegie Mellon University, School of Computer Science, Institute for Software Research International, URL: http://reports-archive.adm.cs.cmu.edu/anon/isri2004/abstracts/04-105.html

[36] Ukkonen, E. (1985). Algorithms for approximate string matching. *Information and Control*, 64, 100-118.

[37] *United States District Court Southern District of Texas, Indictment*. (2002). Retrieved October 8, 2004, from http://news.findlaw.com/hdocs/docs/enron/usandersen030702ind.pdf

[38] Wasserman, S., & Faust, K. (1994). *Social Network Analysis*. New York: Cambridge University Press.

Graph Theoretic and Spectral Analysis of Enron Email Data

Anurat Chapanond, Mukkai S. Krishnamoorthy, Bülent Yener *

Abstract

Analysis of social networks to identify communities and model their evolution has been an active area of recent research. This paper analyzes the Enron email data set to discover structures within the organization. The analysis is based on constructing an email graph and studying its properties with both graph theoretical and spectral analysis techniques. The graph theoretical analysis includes the computation of several graph metrics such as degree distribution, average distance ratio, clustering coefficient and compactness over the email graph. The values of metrics in Enron email graph are compared to those in another email data set. It is shown that the degree distribution of the Enron email graph obeys the power law and there is a giant component that contains 62% of the nodes. The spectral analysis shows that the email adjacency matrix has a rank-2 approximation.

1 Introduction

There has been an increasing research focus on identifying communities within social networks and modeling their evolution over time. Real data for social network analysis can be obtained from email communications, chat-friendship (i.e., buddy list) lists, or from a non-electronic medium such as membership of clubs or board of directors of Fortune-500 companies.

In this paper, we consider the Enron email data set; this is the only substantial collection of real email data set that is public [8]. We provide both graph-theoretic and spectral analysis of the data set to identify and quantify its structural information. Our approach is based on constructing an adjacency matrix representing the email communication graph. We compute interesting graph properties, such as diameter, clustering coefficient and betweenness of the Enron email graph. We compare the graph properties of Enron email graph with the RPI email graph [2]. The comparison shows that there are both similarities (e.g., both degree distributions obey the power law) and differences (e.g., different density of connectivity among communities of practice) between them. We also perform spectral analysis of the email data (as a matrix). We show that this matrix has a low rank-2 approximation.

There has been prior work on Enron data. In [10] authors automate classification of email messages into user-specific folders and extract from chronologically ordered email streams using SVM(Support Vector Machines). In [5] authors construct a database and provide a brief statistical report. In cite[9] language usage in a social network is studied. In [6] email response times are predicted from email logs.

This paper is organized as follows. In Section 2 we explain how to process the email data set to construct an undirected simple graph (i.e., without self loops). In Section 3 we introduce graph metrics and compare their values to that in RPI email graph. Section 4 presents the spectral analysis and show that rank-2 approximation is possible. In Section 5 we display the email graph using a novel visualization tool. Section 5.1 forms the conclusion.

2 Data Pre-Processing

Enron email data are stored in text file format [9]. There were 150 employees from Enron with email logs recorded during a 19 month period (from De-

*Department of Computer Science Rensselaer Polytechnic Institute, Troy, NY 12180, email(chapaa; moorthy; yener)@cs.rpi.edu. This research is supported in part by NSF ITR Award #0324947.

cember 1999 to June 2001). Each log file contains email headers e.g. Message-ID, Date, From, To, Subject and email content. The attachments, although specified by X-Filename, are not included in the log.

2.1 Resolving Multiple Email Address

We extracted the From and To fields of email headers to build sender- and receiver-email list.

However, there could be several email addresses for an employee, thus we first identify all the email addresses of the same person. For example the following email addresses belong to the same person: vince.kaminski@enron.com, vince.j.kaminski@enron.com, vince_j_kaminski@enron.com, j..kaminski@enron.com, kaminski@enron.com, vincent.j.kaminski@enron.com, j'.'kaminski@enron.com, j.kaminski@enron.com.

While some of these email addresses could be identified automatically, manual inspection is necessary to handle the employees with the same last name or unexpected characters in the emails.

2.2 Construction of the Email Graph

A matrix of number of emails that are sent between Enron employees is constructed. The matrix can be used to construct a directed simple graph, in which vertices represent employees and undirected edges are added between employees who corresponded through email [1] . However we constructed an undirected simple graph using the following threshold; the minimum number of emails between each employee and the minimum number of emails sent by each of them.

Choice of Threshold

The undirected email graph is constructed as follows: in order for two employees to be connected by an edge in the graph two criterion must be met:

1. The employees must have exchanged at least 30 emails with each other.
2. Each member of the pair has sent at least 6 emails to the other (to reduce the number of one-way relationships).

[1] We construct an email graph without processing the email content to minimize the privacy concern.

T1	T2				
	0.05	0.10	0.15	0.20	0.25
25	-0.80	-0.83	-0.92	-0.95	-1.05
30	-0.87	-0.89	-1.01	-1.05	-1.15
35	-0.95	-1.06	-1.13	-1.18	-1.30
40	-1.01	-1.09	-1.24	-1.31	-1.41
45	-1.07	-1.20	-1.33	-1.46	-1.52

Table 1: Exponent value of power law degree distribution on different thresholds T1 and T2.

Changing the value used for each criteron will change the structure of the email graph. We found that the degree distribution of the email graph obeys the power law as shown in Figure **??**. We investigate the degree distribution of the email graph constructed by different thresholds.

We found that by varying the threshold we can construct an email graph with varying exponent value of the power law degree distribution. Table 1 shows different exponent value of the power law degree distribution for different thresholds. In the table, T1 is the minimum number of emails between employees and T2 is the minimum percentage of T1 of emails sent by each of them. We chose T1 of 30 emails and T2 of 20% or 6 emails. The resulting graph has the exponent value of -1.05.

We note that in [1] authors also used T1= 30 and T2= 5 emails as threshold values.

3 Structural Analysis with Graph Metrics

In this section we investigate the properties of Enron email graph with respect to some graph metrics and present a comparison to RPI email graph [2].

3.0.1 Graph metrics

The graph metrics we consider in this paper are degree distribution, diameter, average distance, average distance ratio, compactness, clustering coefficient, betweenness, relative interconnectivity, and relative closeness. We compare the values for two different email graphs, namely, Enron email graph with 150 nodes and RPI email graph with 1681 nodes. RPI email data set is constructed from a full SMTP (Simple Mail Transport Protocol) feed at Rensselaer Polytechnic Institutes central mail servers during a 24-hour period on 01/05/2004. Personally identifiable information in the logs was obscured using the HMAC message authentication protocol with a 128bit SSH1 hash as described in [2]. There are two differences between the construction of current RPI email data set and the one used in [2]: (i) the current set excludes the emails from and to outside of RPI domain, and (ii) it is subject to the thresholding as explained above.

Degree distribution - Degree distribution is the histogram of the degree of vertices in the graph. Degree distribution of an email graph reflects the power law property of the graph. It is used to determine an appropriate threshold for constructing the email graph. The degree distribution log graph for Enron and RPI email graph are shown in Figure **??**

Diameter - Diameter is the longest of the shortest paths between any pair of vertices in a connected graph. It reflects how far apart two vertices are (from each other) in the graph. We computed the diameter of the giant component for both the Enron and the RPI email graphs. The Enron graph has a 9 diameter and the RPI graph has 27. The RPI graph has higher value of diameter than the Enron graph because the RPI graph has about ten times more number of vertices. We note that the diameters are surprisingly high in both graphs with respect to the number of vertices.

Average distance ($AvgDist$) - Average distance is the average length of shortest path between each vertex in the graph. The vertices that do not have a shortest path between them will be given the number of vertices in the graph as the length of their shortest path.

Average distance ratio - Average distance ratio is defined as $\frac{NodeNo - AvgDist}{NodeNo}$ where NodeNo is the total number of vertices in the graph. Average distance ratio can have value between 0 and 1. The graph with only isolated vertices will have the average distance ratio of 0 and the complete graph will have the average distance ratio of 1. Average distance ratio reveals the spanning of edges in the graph; the more spanning the graph is the higher the value of average distance ratio. The value for the Enron graph is 0.36 and for the RPI graph is 0.11. This may indicate that the Enron email graph reflects the organizational structure.

Compactness - Compactness is the ratio between the number of existing edges and the number of all possible edges $\frac{2E}{N^2-N}$ where E is the total number of edges and N is the total number of vertices in the graph. Compactness can have value between 0 and 1. The graph with only isolated vertices will have the compactness of 0 and the complete graph will have the compactness of 1. Compactness is the statistic that is not affected by the structure of the graph since only the number of edges is used to compute. The value for Enron graph is 0.0067 and for the RPI graph is 0.0006. We note that the denominator has N^2, therefore the value of compactness is heavily affected by the size of the graph.

Clustering coefficient - Clustering coefficient C_i is defined as the percentage of the connections between the neighbors of vertex i, i.e. $C_i = \frac{2.E_i}{k.(k-1)}$ where k is the number of neighbors of vertex i and E_i is the number of existing connections between its neighbors. Clustering coefficient is the average value of C_i for all vertex i [2]. Clustering coefficient reflects the connectivity information in the neighborhood environment of a vertex. It provides the transitivity information since it controls whether two different vertices are connected or not, assuming that they are connected to the same vertex. The value for the Enron graph is 0.033 and for the RPI graph is 0.119.

Betweenness - The betweenness of an edge is defined as the number of shortest paths that traverse it [1]. The edge with high betweenness is said to be the inter-community edge where the edge with low betweenness is said to be the intra-community edge. By

repeatedly removing an edge with high betweenness the resulting graph will contain a group of clusters where each cluster represents a community of practice [1].

Relative interconnectivity $RI(C_i, C_j)$ between two clusters C_i and C_j is defined as the absolute interconnectivity between C_i and C_j, normalized with respect to the internal interconnectivity of the two clusters C_i and C_j [3].

$$RI(C_i, C_j) = \frac{|EC_{\{C_i,C_j\}}|}{|EC_{C_i}| + |EC_{C_j}|}$$

Where $EC_{\{C_i,C_j\}}$ is the edge-cut of the cluster containing both C_i and C_j so that the cluster is broken into C_i and C_j, and EC_{C_i} (EC_{C_j}) is the size of its min-cut bisector for cluster C_i (C_j).

Relative closeness - $RC(C_i, C_j)$ between a pair of clusters C_i and C_j is the absolute closeness between C_i and C_j, normalized with respect to the internal closeness of the two clusters C_i and C_j [3].

$$RC(C_i, C_j) = \frac{\bar{S}_{EC_{\{C_i,C_j\}}}}{\frac{|C_i|}{|C_i|+|C_j|}\bar{S}_{EC_{C_i}} + \frac{|C_j|}{|C_i|+|C_j|}\bar{S}_{EC_{C_j}}}$$

Where $\bar{S}_{EC_{\{C_i,C_j\}}}$ is the average weight of the edges that connect vertices in C_i to vertices in C_j and $\bar{S}_{EC_{C_i}}$ ($\bar{S}_{EC_{C_j}}$) is the average weight of the edges that belong to the min-cut bisector of cluster C_i (C_j).

Relative interconnectivity and relative closeness are metrics used to determine the similarity in graph structure between two clusters. In this paper we use the metrics to determine the similarity of community of practice in the graph. By the definition of relative closeness, our graph, an undirected simple graph with equal edge weights, will always have the value of 1 for relative closeness of any clusters. The connectivity between each cluster is also of interest. It can be used to analyze the pattern or type of community of practice in the graph.

3.1 Comparison of the Enron and RPI data sets

Table 2 shows the comparison of graph properties and metrics between the Enron and RPI graphs.

Graph properties and metrics	Enron	RPI
Number of vertices	150	1681
Number of edges	52	1932
Number of connected components	57	290
Size of giant component	93	535
Diameter	9	27
Average Distance Ratio	0.36	0.11
Compactness	0.0067	0.0006
Clustering Coefficient	0.033	0.119

Table 2: The comparison of graph properties and metrics for Enron and RPI data.

Different values from the metrics suggested that these two graphs have different organizational structures. We found that the Enron graph has a smaller giant component than RPI graph because of its smaller size. The giant component in the Enron graph contains 62% of the vertices. The Enron graph structure, with higher value of average distance ratio and compactness; seems to be more clustered than the RPI graph . However, the clustering coefficient shows that RPI graph is more clustered. We show in section 3.3 that this conflict can be explained by the analysis of their clusters.

The degree distribution for the Enron and the RPI graph are shown in Figure **??**. These show that both graphs obey power law distribution.

3.2 Graph Clustering

We constructed the communities of practice from the Enron graph by the algorithm described in [1]. The algorithm is a clustering method that repeatedly removes an edge of the graph by betweenness metric until the graph reaches stopping criteria. The edge with highest betweenness will be removed until the component size is less than 6 or all edges in the component has betweenness less than the number of vertices in the component minus one. We then calculated relative interconnectivity between each cluster.

The Enron graph has 27 communities of practice excluding all communities with only one vertex. There are 50 links (relative interconnectivity between

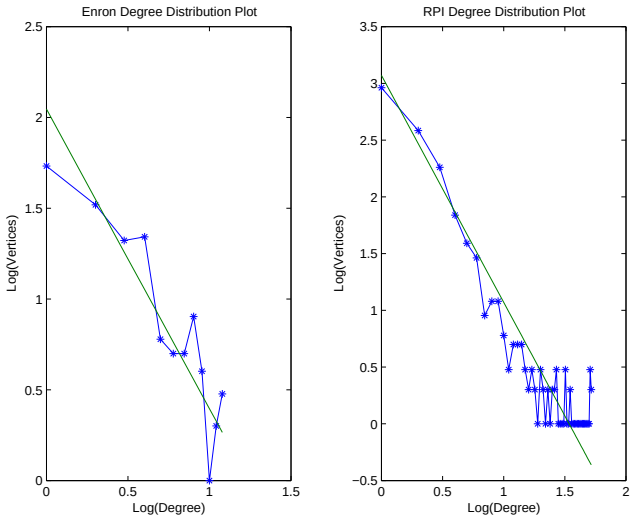


Figure 1: The log-log degree distribution plot for the Enron and the RPI email graph.

two clusters more than zero) between its communities. However the RPI graph has 472 communities of practice and there are only 212 links. We also found many cliques in RPI community of practice. In summary we found that the connectivity inside the communities of practice in Enron graph is sparser than that in the RPI graph but the connectivity between communities of practice in the Enron graph is denser than that in the RPI graph.This explains the conflict when comparing Enron and RPI graph metrics. Enron graph has a sparser connectivity inside the communities which results in a lower value of clustering coefficient but with a denser connectivity between communities Enron graph has a higher value of average distance ratio and compactness. We also found the pattern of the connectivity in RPI graph. We found that all the communities of practice have only a few links to the other communities. This is because the vertices mostly represent students or teachers and they are bound by the number of classes they involve. However Enron graph has different pattern; some communities could have high number of links where some communities have small number of links. Therefore we conclude that we can analyze pattern or type of community from the metric relative interconnectivity.

4 Spectral Analysis of the Enron Data Set

In this section, we perform a spectral analysis on Enron email data similar to what was done with the RPI email data [2]. We show that the Enron email matrix has also a low rank (i.e., rank 2) approximation. This is accomplished by performing Singular Value Decomposition [14] of the Enron email matrix (that was done using the preprocessing steps mentioned in the earlier sections). We also perform a simple clustering of the data based on the low rank approximation.

In matrix notation, SVD for Enron email matrix of $m \times m$ is defined as $A = U\Sigma V^T$ where U and V are orthogonal (thus $U^TU = I$ and $V^TV = I$) matrices of dimensions $m \times r$ and $m \times r$ respectively, containing the left and right singular vectors of A. $\Sigma = \mathbf{diag}(\sigma_1(A), \ldots, \sigma_r(A))$ is an $r \times r$ diagonal matrix containing the singular values of A. SVD has been extensively used in analyzing large data [5]. The plot of the singular values are shown in Figure 4.

The largest two singular values of the Enron email matrix are 2277 and 1550 and the rest of the singular values are much smaller than these two values. So, we claim that Enron email matrix has a low rank (2) approximation. In other words, all the entries in the Enron email matrix can be approximately obtained using two principal components.

Once we obtained that the matrix has a low rank approximation, we projected the matrix in each of the dimensions. Plotting the data in the first dimension, we computed three clusters in the first dimension. The first cluster consisting of indices 20,44,57 and 126, which are Jeffrey Dasovich, Mary Hain, Steven Kean, and James Steffes, the second consisting of indices 1,8,23,43,56,61,63,73,105,109,117 and 133, which are Philip Allen, Sally Beck, David Delainey, Mark Haedicke, Wincente Kaminski, Louise Kitchen, John Lovorato, Kay Mann, Elizabeth Sager, Richard Sanders, Richard Shapiro, and Mark Taylor, and the third cluster containing the rest of

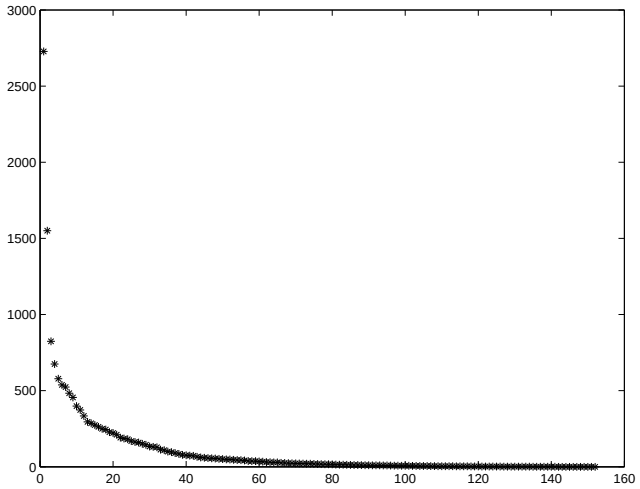


Figure 2: The singular values of Enron email matrix shows that largest two singular values will be sufficient for noise reductions and extracting the structure.

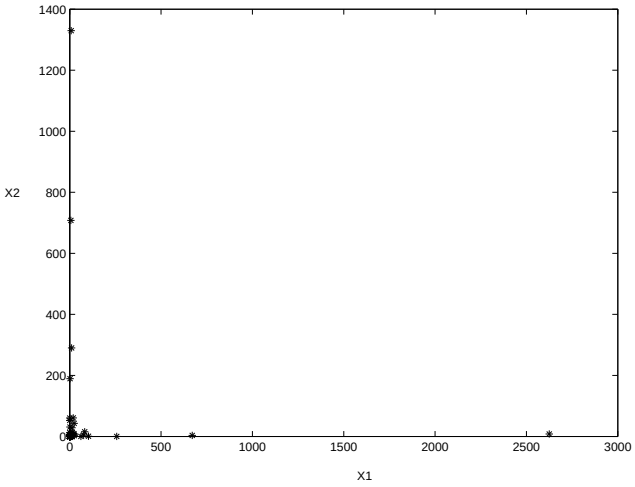


Figure 3: Projection of entries in Rank-2.

the indices. Plotting the data in the second dimension, we computed three clusters. The first cluster consists of indices 55,115,125,135, which are Tana Jones, Sara Shackleton, Carol St Clair, Paul Thomas, the second cluster consisting of indices 8,43,47,54,73,87,90,105,109 , which are Sally Beck, Mark Haedicke, Marie Heard, Kay Mann, Stephanie Panus, Debra Perlingiere, Elizabeth Sager, Richard Sanders and the third cluster containing the rest of the indices. The clusters that are obtained using this are more or less consistent with the clusters that are obtained using the graph model. Finally, we show the actual distribution of the entries of the matrix projected into the two dimension in th next Figure 3

5 Visualization: Email Graph to Organization Hierarchy

The following image 4 shows the visualization of the Enron graph. The layout was done with GraphDraw, a graph tool in Java [13] The visualization is automatically created by using a force-directed algorithm from email graph.

Each vertex will try to push the other vertices away while each edge acts like a spring that pulls the vertices together. The graph has been color-coded by cluster of community of practice. The vertices with the same color are in the same community of practice. The giant connected component of the Enron graph is shown but some isolated vertices are omitted.

Visual inspection of the graph reveals the organization leadership tends to end up in the center. We did not know the hierarchy of the Enron organization however we looked at the highly paid executives [8]. We found that the resulting email graph showed somewhat the hierarchy of the organization.

Using a BFS algorithm a spanning tree with the root of the tree being the vertex corresponding to Enron CEO (level 0). We found that the level of vertices corresponds to the salary of the employee; i.e. the higher payment an employee receives, the lower level (smaller number) the vertex is.

5.1 Summary and Conclusions

In this paper we construct a graph from the Enron email data set and analyze its both graph theoretical and spectral properties. We also compare the En-

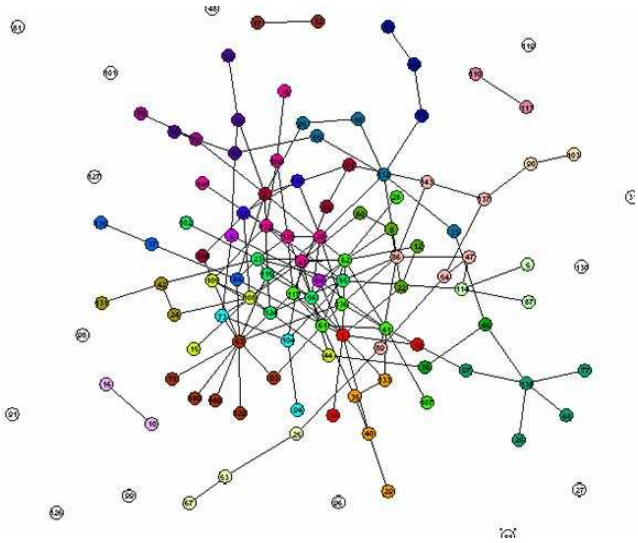


Figure 4: The visualization for Enron email graph color-coded by the cluster of community of practice.

Employee	Payment	Level
Kenneth Lay	$103,559,793.00	0
Philip Allen	$4,484,442.00	1
David Delainey	$4,749,979.00	2
Mark Haedicke	$3,859,065.00	2
Louise Kitchen	$3,471,141.00	2
Rick Buy	$2,355,702.00	2
Wincenty Kaminski	$1,085,821.00	2
Richard Shapiro	$1,057,548.00	2
Mitchell Taylor	$1,092,663.00	2
Sally Beck	$969,068.00	2
John Lavorato	$10,425,757.00	3
Jeffrey Shankman	$3,038,702.00	4
Michael Mcconnell	$2,101,364.00	4
Steven Kean	$1,747,522.00	4
James Derrick	$550,981.00	4
Roderick Hayslett	$0.00	6

Table 3: The payment and spanning tree level for each Enron executives.

ron email graph to the RPI email graph. Some of the observations can be summarized as follows: The degree distribution of the Enron email graph obeys power law and there is a giant component that contains 62% of the vertices. The graph metrics considered for analyzing the properties of email graphs are useful to capture the social structure. For example based on the *betweenness* metric we observe that the connectivity between communities of practice in the Enron email graph is denser than that in the RPI email graph. Furthermore, in the Enron graph some communities have a high number of links while other communities have a small number of links. This is in contrast with the RPI email graph in which communities of practice have only a few links to the other communities. This may be because the vertices mostly represent students or faculty and the communities are related to the classes. Thus the metric *relativeinterconnectivity* can be used to analyze the pattern or type of community.

The visualization of the email graph shows somewhat the hierarchy of the organization with respect to the salary structure.

We also investigate whether there is any significant link between Enron employees and people from White House. We add a vertex that represents people from White house, e.g. president@whitehouse.gov, vice.president@whitehouse.gov. Our preliminary investigation shows that there are emails being sent and received between Enron employees and White House during the logging period but after the filtering process there is no link between this group of Enron employees and White House people. We also examined the link between Enron employees and the six people who had been prosecuted - Sheila Kahanek, Dan Boyle, Daniel Bayly, Robert Furst, William Fuhs, and James Brown. By adding another vertex representing these people we found that there is no link between them and this group of Enron employees.

Acknowledgments: The authors would like to thank Michael D. Sofka for providing the RPI email data set.

References

[1] Tyler, J. R., Wilkinson, M. D., and Huberman. B. A., "Email as Spectroscopy: Automated Discovery of Community Structure within Organizations", in Pro-

ceeding of the International Conference on Communities and Technologies, Netherlands, kluwer Academic Publishers (2003).

[2] Drineas, P., Krishnamoorthy, M. S., Sofka, M. D., and Yener, B., "Studying E-mail Graphs for Intelligence Monitoring and Analysis in the Absence of Semantic Information", 2004.

[3] Karypis, G., Han, E.-H., and Kumar, V., "CHAMELEON: A hierarchical clustering algorithm of spatial data", In Proc. 8th Symp. Spatial Data Handling, pages 45-55, Vancouver, Canada, 1998.

[4] Chapanond, A., and Krishnamoorthy M. S., "User Classification for P2P network", manuscript (2004).

[5] Han, J., and Kamber, M., Data Mining : Concepts and Techniques, Morgan Kaufmann Publishers, 2001.

[6] Adibi, J., and Shetty, J., The Enron Email Dataset Database Schema and Brief Statistical Report, `http://www.isi.edu/~adibi/Enron/Enron_Dataset_Report.pdf`.

[7] Kalman, Y., Rafacli, S., Email Chronemics: Unobtrusive Profiling of Response Times, HICSS-38, Hawaii, 2005.

[8] Houston Chronicles, `http://www.chron.com/content/chronicle/special/01/enron/index.html`.

[9] Enron Email Dataset, `http://www-2.cs.cmu.edu/~enron/`.

[10] Corrada-Emmanuel, A., McCallum, A., and Wang, X., Language Use in a Social Network: The Enron Email Dataset, CNLP Seminars, 2004.

[11] Klimt, B., and Yang, Y., The Enron Corpus: A New Dataset for Email Classification Research, To be published in proceedings of the European Conference on Machine Learning (ECML), 2004.

[12] Loch, C. H., Tyler, J. R., and Lukose, R., "Conversational Structure in Email and Face to Face communication", Draft, submitted to Organization Science.

[13] N. Preston and M. Krishnamoorthy, "GraphDraw-A Graph Drawing System to study Social Networks," Unpublished Manuscript, Rensselaer Polytechnic Institute, Troy, NY, 2004.

[14] G. Golub and F. Van Loan, "Matrix Computations", Johns Hopkins University Press, 1984.

Scan Statistics on Enron Graphs*

Carey E. Priebe† John M. Conroy‡ David J. Marchette§ Youngser Park†

Abstract

We introduce a theory of scan statistics on graphs and apply the ideas to the problem of anomaly detection in a time series of Enron email graphs.

1 Introduction.

Consider a directed graph (digraph) D with vertex set $V(D)$ and arc set $A(D)$ of directed edges. For instance, we may think of D as a communications or social network, where the $n = |V(D)|$ vertices represent people or computers or more general entities and an arc $(v, w) \in A(D)$ from vertex v to vertex w is to be interpreted as meaning "the entity represented by vertex v is in directed communication with or has a directed relationship with the entity represented by vertex w." We are interested in testing the null hypothesis of "homogeneity" against alternatives suggesting "local subregions of excessive activity." Toward this end, we develop and apply a theory of scan statistics on random graphs.

2 Scan Statistics.

Scan statistics are commonly used to investigate an instantiation of a random field X (a spatial point pattern, perhaps, or an image of pixel values) for the possible presence of a local signal. Known in the engineering literature as "moving window analysis", the idea is to scan a small window over the data, calculating some local statistic (number of events for a point pattern, perhaps, or average pixel value for an image) for each window. The supremum or maximum of these locality statistics is known as the scan statistic, denoted $M(X)$. Under some specified "homogeneity" null hypothesis H_0 on X (Poisson point process, perhaps, or Gaussian random field) the approach entails specification of a critical value c_α such that $P_{H_0}[M(X) \geq c_\alpha] = \alpha$. If the maximum observed locality statistic is larger than or equal to c_α, then the inference can be made that there exists a nonhomogeneity — a local region with statistically significant signal.

*Corresponding author: Carey E. Priebe = <cep@jhu.edu>
†Johns Hopkins University, Baltimore, MD
‡IDA Center for Computing Sciences, Bowie, MD
§NSWC B10, Dahlgren, VA

An intuitive approach to testing these hypotheses involves the partitioning of the region X into disjoint subregions. For cluster detection in spatial point processes this dates to Fisher's 1922 "quadrat counts" [7]; see [6]. Absent prior knowledge of the location and geometry of potential nonhomogeneities, this approach can have poor power characteristics.

Analysis of the univariate scan process ($d = 1$) has been considered by many authors, including [10], [3], [4], and [8]. For a few simple random field models exact $p-$values are available; many applications require approximations to the $p-$value. The generalization to spatial scan statistics is considered in [10], [1], [8], and [2]. As noted by [5], exact results for $d = 2$ have proved elusive; approximations to the $p-$value based on extreme value theory are in general all that is available. [9] present an alternative approach, using importance sampling, to this problem of $p-$value approximation.

3 Scan Statistics on Graphs.

The order of the digraph, $n = |V(D)|$, is the number of vertices. The size of the digraph, $|A(D)|$, is the number of arcs. For $v, w \in V(D)$ the digraph distance $d(v, w)$ is defined to be the minimum directed path length from v to w in D.

For non-negative integer k (the *scale*) and vertex $v \in V(D)$ (the *location*), consider the closed kth-order neighborhood of v in D, denoted $N_k[v; D] = \{w \in V(D) : d(v, w) \leq k\}$. We define the *scan region* to be the induced subdigraph thereof, denoted

$$\Omega(N_k[v; D]),$$

with vertices $V(\Omega(N_k[v; D])) = N_k[v; D]$ and arcs $A(\Omega(N_k[v; D])) = \{(v, w) \in A(D) : v, w \in N_k[v; D]\}$. A *locality statistic* at location v and scale k is any specified digraph invariant $\Psi_k(v)$ of the scan region $\Omega(N_k[v; D])$. For concreteness consider for instance the *size* invariant, $\Psi_k(v) = |A(\Omega(N_k[v; D]))|$. Notice, however, that any digraph invariant (e.g. density, domination number, etc.) may be employed as the locality statistic, as dictated by application. The "scale-specific" *scan statistic* $M_k(D)$ is given by some function of the collection of locality statistics $\{\Psi_k(v)\}_{v \in V(D)}$; consider for instance

the maximum locality statistic over all vertices,

$$M_k(D) = \max_{v \in V(D)} \Psi_k(v).$$

This idea is introduced in [11].

Under a null model for the random digraph D (for instance, the Erdos-Renyi random digraph model) the variation of $\Psi_k(v)$ can be characterized and $M_k(D)$ large indicates the existence of an induced subdigraph (scan region) $\Omega(N_k[v; D])$ with excessive activity. A test can be constructed for a specific alternative of interest concerning the structure of the excessive activity anticipated. However, if the anticipated alternative is, more generally, some form of "chatter" in which one (small) subset of vertices communicate amongst themselves (in either a structured or an unstructured manner) then our scan statistic approach promises more power than other approaches.

Finally, we wish to consider the scan statistic which accounts for variable scale. Let $K \subset \{1, \cdots, n-1\}$ be a collection of scales, and let Ψ'_k be a scale-standardized version of the locality statistic Ψ_k. For instance, for given $\alpha \in (0, 1)$, find $g_{k,\alpha}(\cdot)$ such that $\Psi'_k(v) = g_{k,\alpha}(\Psi_k(v))$ satisfies $P[\Psi'_k(v) \geq c_\alpha] \approx \alpha$ for all $v \in V(D)$ and for all $k \in K$. This standardization imposes upon each locality statistic the same probability of exceedance. Then the *scan statistic* $M_K(D)$ is given by

$$M_K(D) = \max_{k \in K} \max_{v \in V(D)} \Psi'_k(v)$$

and we reject for large values of $M_K(D)$.

For the Enron data considered in this paper, as for much social network data, no appropriate simple null random graph model is obvious. The dataset, as we process it, consist of a time series of digraphs $D_1, D_2, \cdots, D_{T=189}$. We will proceed conditionally: we will assume that the data (or the statistics derived from the data) have some short-time stationarity properties under the null, so that a moving window approach is appropriate. We will be concerned with discovering anomalies that appear as digraphs which differ substantially from those seen in the recent past. In particular, we wish to detect subdigraphs with an unusually high connectivity, as measured by our statistic. This conditional approach alleviates the requirement to posit an appropriate and simple null graph model — but does require some (approximate) stationarity.

4 The Enron Data.

The Enron email dataset is available online [12]. This dataset consists of a collection of 150 folders corresponding to the email to and from senior management and others at Enron, collected over a period from about 1998 to 2002. The emails have been minimally processed to correct integrity problems. Some emails have been deleted, as have all attachments. Thus, while imperfect, this dataset represents a rich environment in which to perform text analysis and link analysis. More information on this dataset can be found online [13].

One consequence of the processing of these data is that some of the original email addresses have been changed. Invalid addresses were converted to *no_address@enron.com*. In several cases, individuals have multiple addresses, which are clearly a result of some post-processing: for example, Phillip K. Allen has email addresses *phillip.allen@enron.com* and *k..allen@enron.com*. In this study we will treat such cases as distinct; one potential goal might be to recognize this "aliasing" from the link analysis alone, without reference to the content of the messages. This will be discussed further in Section 7.1.

5 Whence Our Enron Graphs?

The data are collected from "about 150 users" — mostly Enron executives, but also some energy traders, executive assistants, etc. However, our graphs are based on 184 users, which is the number of unique addresses we obtain from the 'From' line of emails in the 'Sent' boxes after manually removing some addresses which are clearly not associated with the 150 users. (NB: Neither of the two extreme options — keeping all addresses, or merging to the point of one-to-one correspondence between addresses and known users — seems practical; the former yields too many obvious aliases and extraneous addresses, and no simple unassailable version of the latter presents itself to us. Thus, we proceed with an admittedly imperfect collection of vertices.) In addition, some of the time stamps in the original data are clearly invalid, occurring before Enron existed, so we restrict our attention to a period of 189 weeks, from 1998 through 2002.

For each week $t = 1, \cdots, 189$, there is a digraph $D_t = (V, A_t)$ with $|V| = 184$ vertices and directed edges (arcs) A_t, where $(v, w) \in A_t \iff$ vertex v sends at least one e-mail to vertex w during the t-th week. We make no distinction between emails sent "To", "CC" or "BCC".

6 Statistics and Time Series.

Our time-dependent scale-k locality statistic is given by

$$\Psi_{k,t}(v) = |A(\Omega(N_k[v; D_t]))|$$

for $k \in \{1, 2, \cdots, K\}$. In an abuse of notation, we will let $\Psi_{0,t}(v) = \text{outdegree}(v; D_t)$.

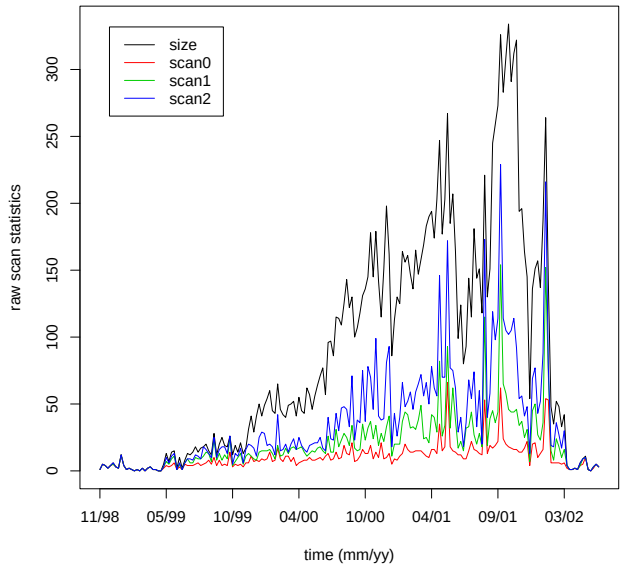


Figure 1: Time series of scan statistics and max degree ($M_{k,t}$ for $k = 0, 1, 2$), as well as digraph size, for weekly Enron email digraphs during a period of 189 weeks from 1998–2002. (See also Figures 8–11.)

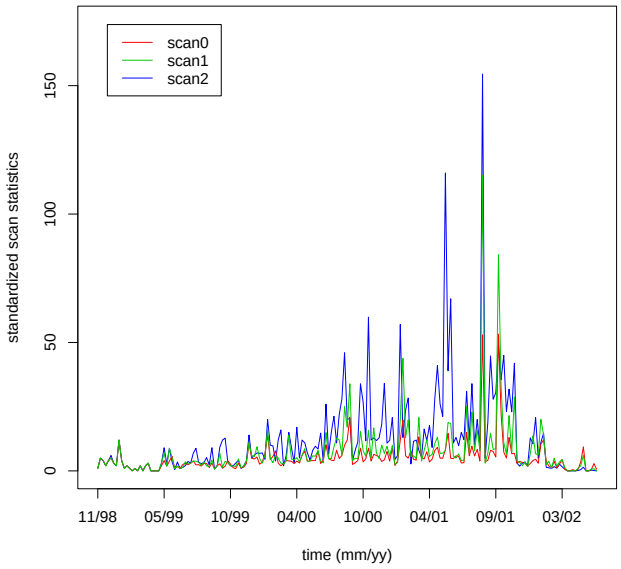


Figure 2: Time series of standardized scan statistics and max degree ($\widetilde{M}_{k,t}$ for $k = 0, 1, 2$) for weekly Enron email digraphs during a period of 189 weeks from 1998–2002. (See also Figures 12–14.)

Figure 1 shows the three statistics

$$M_{k,t} = \max_v \Psi_{k,t}(v); k = 0, 1, 2$$

as well as $size(D_t)$, as functions of time (weeks) $t = 1, \cdots, 189$ for the 189 weeks under consideration. (Figures 8–11 show these four curves separately.)

The raw locality statistics $\Psi_{k,t}(v)$ are inadequate for our purposes. Consider, for instance, the situation in which one vertex, v, has a lot of activity throughout time, and another vertex, w, has but one tenth this amount of activity until one week in which w triples its activity. Without some form of vertex-dependent standardization, the increase in activity for w will go unnoticed, as $v = \arg\max \Psi_{k,t}(v)$ regardless of w's increased activity. Thus the locality statistics $\Psi_{k,t}(v)$ must be standardized using vertex-dependent recent history.

Our vertex-standardized locality statistic, for $k = 0, 1, 2$, is given by

$$\widetilde{\Psi}_{k,t}(v) = \left(\Psi_{k,t}(v) - \widehat{\mu}_{k,t,\tau}(v)\right) / \max(\widehat{\sigma}_{k,t,\tau}(v), 1)$$

where

$$\widehat{\mu}_{k,t,\tau}(v) = \frac{1}{\tau} \sum_{t'=t-\tau}^{t-1} \Psi_{k,t'}(v)$$

and

$$\widehat{\sigma}_{k,t,\tau}(v) = \frac{1}{\tau - 1} \sum_{t'=t-\tau}^{t-1} \left(\Psi_{k,t'}(v) - \widehat{\mu}_{t,\tau}(v)\right)^2.$$

That is, we standardize the locality statistic $\Psi_{k,t}(v)$ by a vertex-dependent mean and standard deviation based on recent history. (The denominator in $\widetilde{\Psi}_{k,t}(v)$ is forced to be greater than or equal to one to eliminate fragility due to vertices with little or no variation in activity.)

In Figure 2 we plot the standardized scan statistics

$$\widetilde{M}_{k,t} = \max_v \widetilde{\Psi}_{k,t}(v)$$

against t over the 189 weeks. (Figures 12–14 show these three curves separately.)

This approach requires a vertex-dependent local stationarity assumption. The validity of a stationarity assumption is obviously suspect over the entire 189 weeks, but short-time near-stationarity (we use $\tau = 20$) may be reasonable as a null model.

7 Anomaly Detection.

Given the standardized scan statistic time series $\widetilde{M}_{k,t}$ presented in Figure 2, we now consider anomaly detection.

For simplicity, we consider a temporally-normalized version of $\widetilde{M}_{k,t}$,

$$S_{k,t} = (\widetilde{M}_{k,t} - \widehat{\mu}_{k,t,\ell})/\max(\widehat{\sigma}_{k,t,\ell}, 1),$$

where $\widehat{\mu}_{k,t,\ell}$ and $\widehat{\sigma}_{k,t,\ell}$ are the running mean and standard deviation estimates of $\widetilde{M}_{k,t}$ based on the most recent ℓ time steps. (Here we use $\ell = 20$.) Detections are defined here as weeks for which $\widetilde{M}_{k,t}$ achieves a value greater than five standard deviations above its mean; i.e., times t such that $S_{k,t} > 5$

Figure 3 depicts $S_{2,t}$ for a 20 week period from February 2001 through June 2001. We observe that the second order scan statistic indicates a clear anomaly at $t^* = 132$ ($\max_v \widetilde{\Psi}_{2,132}(v)$ is a seven sigma event) in May 2001. This anomaly is apparent, in hindsight, in Figure 2.

Inference performed using simple sigmages is inadequate in this case, of course, because there is no reason to believe that the distribution of $S_{k,t}$ is normal or that $S_{k,t}$ and $S_{k,t'}$ are independent. Computational methods such as the bootstrap would be appropriate. We consider exceedance probabilities of an extreme value distribution, the Gumbel, fit via the method of moments. $S_{2,132} = 7.3$; 7.3 standard deviations yields a $p-$value $< 10^{-10}$, assuming normality. While the significance for the detection at $t^* = 132$ is not so drastic under the more reasonable Gumbel model, we nevertheless obtain an exceedance probability $< 10^{-6}$, which remains convincing. Bonferonni analysis suggests that if the $\widetilde{\Psi}_{k,t}$ are approximately distributed as a t_{19} then the detection is significant; however, if the distribution of the $\widetilde{\Psi}_{k,t}$ has extraordinarily heavy tails (e.g., Cauchy) then the $\alpha = 0.05$ level critical value may be greater than 7.3. Thus, under a reasonable range of null distributions, the detection at $t^* = 132$ is statistically significant.

Figure 4 shows the graph topology, sans isolates, for our 'detection' graph D_{132}. Our vertex of interest, $v^* = \arg\max_v \widetilde{\Psi}_{2,132}(v)$, is identified with email address *k..allen*. Of note is the fact that $\arg\max_v \Psi_{0,132}(v) =$ *john.lavorato*. That is, the vertex of maximum outdegree for $t^* = 132$ is *not* the cause of our detection. Furthermore, $\arg\max_v \Psi_{1,132}(v) =$ *john.lavorato*, $\arg\max_v \Psi_{2,132}(v) =$ *richard.shapiro*, $\arg\max_v \widetilde{\Psi}_{0,132}(v) =$ *richard.shapiro*, and $\arg\max_v \widetilde{\Psi}_{1,132}(v) =$ *joannie.williamson*. Thus the detection based on $v^* =$ *k..allen* is apparent only when using the standardized second order scan statistic.

Table 1 gives some relevant numerical values for the 'detection' graph D_{132}.

There is excessive activity among the elements of the closed 2-neighborhood of our vertex of interest v^*

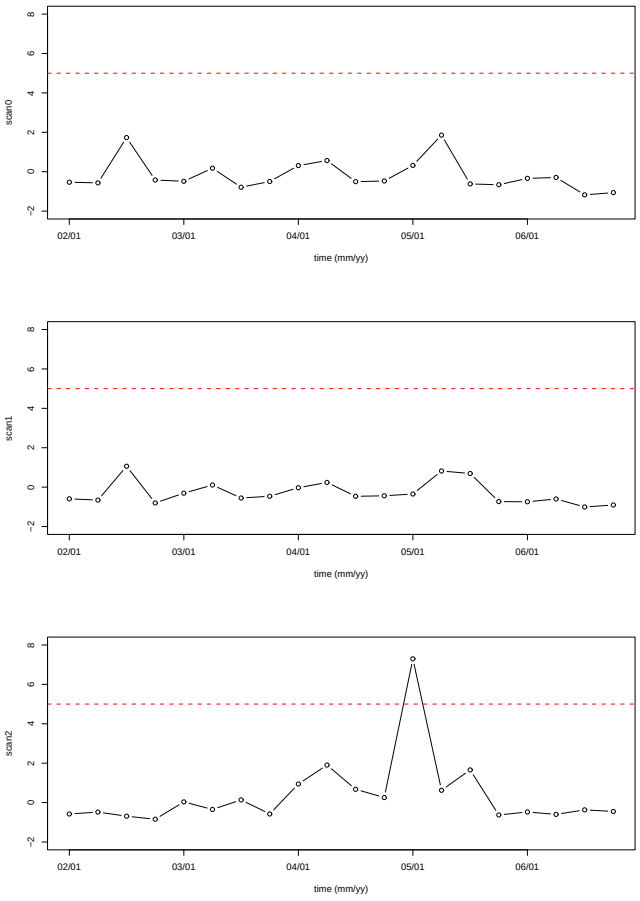


Figure 3: $S_{k,t}$, the temporally-normalized standardized scan statistics, on zoomed-in time series of Enron e-mail graphs during a period of 20 weeks in 2001. Top: $k = 0$; Middle: $k = 1$; Bottom: $k = 2$. This figure shows a detection (a standardized statistic $\widetilde{M}_{k,t}$ which achieves a value greater than 5 standard deviations above its running mean, or a temporally-normalized standardized statistic $S_{k,t}$ in this plot taking a value greater than 5) at week $t^* = 132$ in May 2001 for scale $k = 2$, but not for $k = 1$ or $k = 0$.

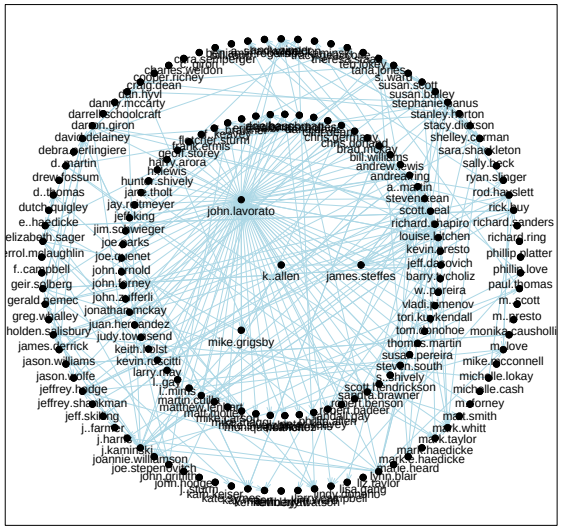


Figure 4: Plot of the 'detection' Enron email graph D_{132} (sans isolates) for which our scan statistic methodology detects an anomaly. The center vertex, *k..allen*, is $v^* = \arg\max_v \widetilde{\Psi}_{2,132}$.

which is not accounted for by its outdegree (or its closed 1-neighborhood). In fact, v^* communicates, in particular, with other vertices each of which has high outdegree. This type of excessive local activity is precisely the raison d'etre for our scan statistics; our approach exhibits the ability to detect this anomaly.

Is this detection an event of interest? It is statistically significant, but the objective of our scan statistic methodology is to sift through massive communications data to find potentially informative events for the purpose of directing additional, more time consuming investigations. The ultimate determination of the practical significance of this or any detection must be made on the basis of subsequent analysis. There is a coinciding insider trading event on the Enron time line ... but there are many insider trading events on the Enron time line! Ideally, one would hope to find a link between the detected excess activity and that insider trading. Such a forensic analysis will require delving into the content of the email messages and associated meta-data.

Time $t^* = 132$ is the only week among the 189 under consideration for which $S_{2,t} \geq 5$. Detections for the other scan statistics — orders 0, 1, and 3 — that may be worth pursuing are summarized here. For maximum standardized outdegree, there are three weeks

time t^*	132 (week of May 17, 2001)		
$size(D_{132})$	267		
scale k	$M_{k,132}$	$\widetilde{M}_{k,132}$	$S_{k,132}$
0	66	8.3	0.32
1	93	7.8	−0.35
2	172	116.0	7.30
3	219	174.0	5.20
number of isolates	50		

Table 1: Details for the 'detection' graph D_{132}.

with $S_{0,t} \geq 5$: 58, 96, 146; for the standardized first order scan statistic, we obtain (almost) the same three detections: 58, 94, 146. The standardized third order scan statistic produces detections at $t^* = 132$ and at week 87.

7.1 Aliasing. In the case of the detection at $t^* = 132$,

$$v^* = \arg\max_v \widetilde{\Psi}_{2,132}(v) = k..allen,$$

perusal of the emails shows that *k..allen* and *phillip.allen* are really the same person. User *k..allen* had no activity before $t^* = 132$, at which time *phillip.allen* switched to the *k..allen* identifier. Thus we have detected an instance of aliasing, which could perhaps have been addressed during the manual merging stage wherein we settled on the collection of 184 vertices to consider. Of course, this identification does in fact require perusal of the emails, which perusal was suggested by the detection ... precisely the point of the exercise!

However, it may be possible to automatically identify such aliasing events. Given the detection (v^*, t^*) we can immediately identify *k..allen* as having had no activity prior to $t^* = 132$. From this point, we may employ a "matched filter" scheme to determine candidates for aliasing by matching the pattern of *k..allen*'s activity at or after $t^* = 132$ against the pattern of other vertices' activity prior to $t^* = 132$. Vertices with a high score for some matching function will be deemed likely candidates for further investigation.

For instance, we may compute, for each vertex $v \in V \setminus \{v^*\}$, the simple score

$$s_{t^*,\kappa}(v; v^*) = \sum_{t'=t^*-\kappa}^{t^*-1} |N_1(v; D_{t'}) \cap N_1(v^*; D_{t^*})|.$$

In this case we obtain *phillip.allen* $= \arg\max_v s_{t^*,\kappa}(v; v^*)$. That is, for this simple case, the aliasing can be automatically identified and resolved.

This idea of employing matched filters to time series of graphs, introduced here in a very simplistic fashion, will be pursued in more detail elsewhere.

7.2 Another Detection. The detection of $v^* = \arg\max_v \widetilde{\Psi}_{2,132}(v) = k..allen$ at $t^* = 132$, while real and interesting, is due to the fact that *k..allen* had not been active prior to $t^* = 132$. We may be interested, instead, in detections for which activity increases from a non-zero baseline. That is, we consider the statistic

$$\widetilde{\Psi}_{k,t}(v) \cdot I\{\widehat{\mu}_{0,t,\tau}(v) > c\},$$

where $I\{E\}$ is the indicator function taking value one if event E occurs and taking value zero otherwise, which requires there to have been some recent activity.

For $c = 1$, one such detection of this type, for which the order $k = 2$ scan statistic detects but the order $k = 0$ and $k = 1$ scan statistics do not detect, is $v^* = rod.hayslett$ at $t^* = 152$ (the week of October 4, 2001).

Table 2 gives the scan statistics for this detection for the weeks up to and including t^*. Here we see clearly the increase in activity, and we see that it is not due to order 0 or order 1 locality statistics. (N.B. It does appear that a detection at $t^* - 2$ may be appropriate.)

scale k	$\Psi_{k,t^*-5:t^*}(v^*)$
0	[1 , 2 , 1 , 3 , 1 , 2]
1	[1 , 2 , 2 , 9 , 2 , 4]
2	[1 , 2 , 2 , 19 , 4 , 175]
3	[1 , 2 , 2 , 58 , 6 , 268]

Table 2: Locality statistics $\Psi_{k,t}(v^* = rod.hayslett)$ for the time range $\{t^* - 5, \cdots, t^*\}$ leading up to the $v^* = rod.hayslett$ detection at $t^* = 152$.

However, further investigation indicates that this detection is due to the fact that *rod.hayslett* communicates with *sally.beck*, and *sally.beck* is an order 0 locality statistic detection at $t^* = 152$ due to a massive increase in outdegree (see Table 3).

Thus, in some sense, neither the *k..allen* / *phillip.allen* detection at $t^* = 132$ nor the *rod.hayslett* / *sally.beck* detection at $t^* = 152$ is really due to the type of excessive "chatter" in which we are most interested.

7.3 Detecting Chatter. For each time t and vertex v, consider the order 2 statistic

$$\widetilde{\Psi}'_t(v) = \left(\widetilde{\Psi}_{2,t}(v) \cdot \mathcal{I}_{t,\tau}(v)\right) / \max(\gamma_t(v), 1).$$

scale k	$\Psi_{k,t^*-5:t^*}(v)$
0	[3 , 2 , 0 , 2 , 3 , 62]
1	[3 , 3 , 0 , 3 , 6 , 154]
2	[4 , 3 , 0 , 37 , 11 , 229]
3	[4 , 3 , 0 , 98 , 16 , 267]

Table 3: Locality statistics $\Psi_{k,t}(v = sally.beck)$ for the time range $\{t^* - 5, \cdots, t^*\}$ leading up to the $v^* = rod.hayslett$ detection at $t^* = 152$.

Here the term $\mathcal{I}_{t,\tau}(v)$ is the product of three indicator functions,

$$I\{\widehat{\mu}_{0,t,\tau} > c_1\},$$

$$I\{\Psi_0(v) < \widehat{\sigma}_{0,t,\tau}(v)c_2 + \widehat{\mu}_{0,t,\tau}(v)\},$$

and

$$I\{\Psi_1(v) < \widehat{\sigma}_{1,t,\tau}(v)c_3 + \widehat{\mu}_{1,t,\tau}(v)\}.$$

That is, we gate the second order scan statistic so that some minimal level of recent activity is required, and we insist that the order 0 and order 1 scan statistics do not yield detections. In this way we narrow the class of alternatives under consideration — the types of anomalous activities that will be deemed detections; we seek a detection in which the excess activity is due to chatter amongst the 2-neighbors. We include an "inhomogeneity penalty" $\gamma_t(v)$, the standard deviation of the outdegrees of the neighbors $N_1(v^*; D_{t^*})$, in the denominator of $\widetilde{\Psi}'_t(v)$ to further narrow our search to the case of "balanced chatter" (and to rule out events such as the *rod.hayslett* / *sally.beck* detection at $t^* = 152$).

The $\arg\max_{(v,t)} \widetilde{\Psi}'_t(v)$ is given by $(v^*, t^*) = (steven.kean, 109)$. (The value of $t^* = 109$ corresponds to the week of December 7, 2000.) Figure 5 displays $\widetilde{M}'_t = \max_v \widetilde{\Psi}'_t(v)$ as well as the temporally-normalized version S'_t.

The raw locality statistics $\Psi_{k,t}(v^*)$ for the time range $\{t^* - 5, \cdots, t^*\}$ leading up to this detection are given in Table 4. As can be seen from Table 4, the raw locality statistics for $k = 0$ and $k = 1$ do not have a substantial signal at $t^* = 109$, while for $k = 2$ the presence of an anomaly is clear.

The inhomogeneity penalty for this detection is $\gamma_{t^*}(v^*) \approx 1.7$; the outdegrees of the five neighbors of $v^* = steven.kean$ are 6,6,6,7,10.

The induced subdigraph at $t^* = 109$, $\Omega(N_2[v^*; D_{t^*}])$, is depicted in Figure 6. We see that $v^* = steven.kean$ has five neighbors, each of which has outdegree between six and ten. That is, this detection is due to v^* communicating with a moderate subset of vertices, each of whom communicates with

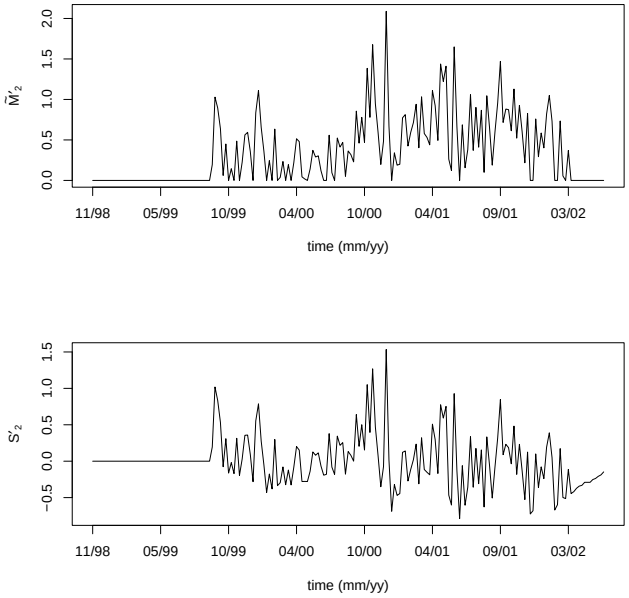


Figure 5: Plot of order 2 statistics $\widetilde{M}'_t$ and S'_t showing the maximum at $t^* = 109$ in December 2000. This is the *steven.kean* "excessive chatter" detection.

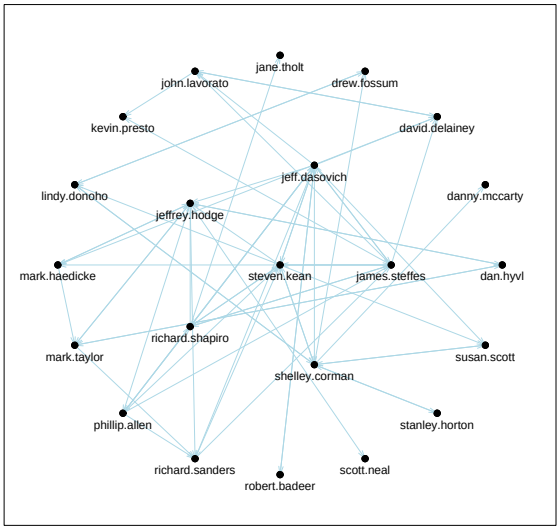


Figure 6: Plot of the 'detection' Enron email graph $\Omega(N_2[v^* = \textit{steven.kean}; D_{t^*=109}])$.

scale k	$\Psi_{k,t^*-5:t^*}(v^*)$
0	[3 , 5 , 4 , 5 , 4 , 5]
1	[11 , 13 , 10 , 10 , 11 , 18]
2	[14 , 35 , 21 , 38 , 13 , 65]

Table 4: Locality statistics $\Psi_{k,t}(v^*)$ for the time range $\{t^*-5, \cdots, t^*\}$ leading up to the *steven.kean* detection at $t^* = 109$.

another moderate subset. Comparing this graph with *steven.kean*'s induced subdigraph $\Omega(N_2[v^*; D_{t^*-1}])$ at $t^* - 1 = 108$ (black arcs and associated vertices in Figure 7) gives a clear, albeit simplistic, indication that change has occurred. Figure 7 gives additional information regarding this change, depicting the subdigraph induced at $t^* - 1 = 108$ by the union of *steven.kean*'s 2-neighborhood at $t^* - 1 = 108$ and *steven.kean*'s 2-neighborhood at $t^* = 109$. The arcs corresponding to communications between members of *steven.kean*'s closed 2-neighborhood at $t^* - 1 = 108$ are depicted in black; gray arcs represent other communications in D_{108} between vertices in *steven.kean*'s 2-neighborhood at $t^* = 109$. Figure 7 shows that this detection is not the result of a simple increase in the size of v^*'s neighborhood, but that the vertices in the neighborhood at t^*, while active at $t^* - 1$, have also increased their activity. Thus, the detection is not due solely to v^* joining a larger group; in addition, the group itself is more active as well. We interpret this figure as suggesting that this detection is robust — insensitive to small changes in the graph.

8 Discussion.

A theory of scan statistics on graphs offers promise for detecting anomalies in time series of graphs.

We have employed perhaps overly-simplistic time series and inference methods, for purposes of illustration; more elaborate methods such as exponential smoothing, detrending, and variance stabilization may be appropriate. In addition, multivariate time series (one time series for each vertex v, in this case) have a theory all their own — e.g., vector autoregressive models — which we have ignored here. And, of course, for data such as this Enron corpus, robust versions of moment estimates we have employed are called for.

Nevertheless, despite our simplistic approach to these various issues, we have demonstrated the potential utility of the scan statistic approach to the problem of anomaly detection in a time series of Enron email graphs. Much remains to be done — mathematically, computationally, and with respect to data and meta-

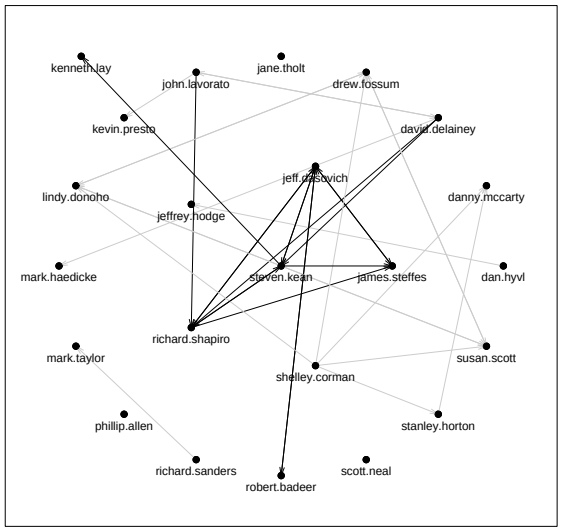


Figure 7: An induced subgraph of D_{108}. Black arcs and associated vertices represent *steven.kean*'s induced subdigraph $\Omega(N_2[v^*; D_{t^*-1}])$ at $t^* - 1 = 108$. Gray arcs represent other communications in D_{108} between vertices in *steven.kean*'s 2-neighborhood at $t^* = 109$. Comparing this figure with Figure 6 provides information regarding the change from $t^* - 1 = 108$ to $t^* = 109$ for the $(v^* = steven.kean, t^* = 109)$ detection.

data analysis. Of particular interest is the extension of these scan statistics to weighted graphs (and hypergraphs), allowing for the detection of anomalies related to the number (and possibly type) of messages sent, as opposed to the simpler case considered herein.

Noteworthy as a closing fact is that the procedures introduced herein can all be performed in a real-time, streaming data environment. That is, a sliding one-week window, rather than disjoint one-week windows, can be utilized and nothing presented herein causes a common laptop computer difficulty in keeping up. Thus, these procedures can be applied in scenarios of online analysis, in addition to the forensic scenario offered by this Enron corpus.

References

[1] R.J. Adler, *The Supremum of a Particular Gaussian Field*, Annals of Probability, 12 (1984), pp. 436–444.

[2] J. Chen and J. Glaz, *Two-Dimensional Discrete Scan Statistics*, Statistics and Probability Letters, 31 (1996), pp. 59–68.

[3] N.A.C. Cressie, *On Some Properties of the Scan Statistic on the Circle and the Line*, Journal of Applied Probability, 14 (1977), pp. 272–283.

[4] N.A.C. Cressie, *The Asymptotic Distribution of the Scan Statistic under Uniformity*, Annals of Probability, 8 (1980), pp. 828–840.

[5] N.A.C. Cressie, *Statistics for Spatial Data*, John Wiley, New York, 1993.

[6] P.J. Diggle, *Statistical Analysis of Spatial Point Patterns*, Academic Press, New York, 1983.

[7] R.A. Fisher, H.G. Thornton, and W.A. Mackenzie, *The Accuracy of the Plating Method of Estimating the Density of Bacterial Populations, with Particular Reference to the Use of Thornton's Agar Medium with Soil Samples*, Annals of Applied Biology, 9 (1922), pp. 325–359.

[8] C.R. Loader, *Large-Deviation Approximations to the Distribution of Scan Statistics*, Advances in Applied Probability, 23 (1991), pp. 751–771.

[9] D.Q. Naiman and C.E. Priebe, *Computing Scan Statistic p–Values using Importance Sampling, with Applications to Genetics and Medical Image Analysis*, Journal of Computational and Graphical Statistics, 10 (2001), pp. 296–328.

[10] J.I. Naus, *Clustering of Random Points in Two Dimensions*, Biometrika, 52 (1965), pp. 263–267.

[11] C.E. Priebe, *Scan Statistics on Graphs*, Technical Report No. 650, Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD 21218–2682, (2004).

[12] `www.cs.queensu.ca/home/skill/siamworkshop.html`

[13] `www-2.cs.cmu.edu/~enron`

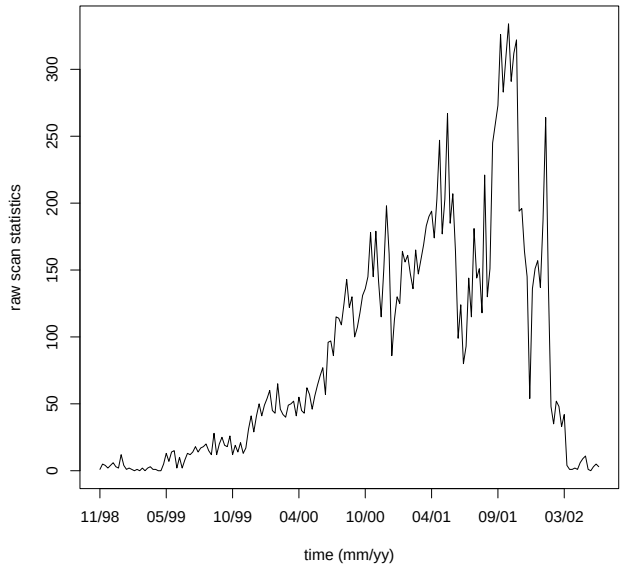


Figure 8: Time series of digraph size for weekly Enron email digraphs during a period of 189 weeks from 1998–2002. (See also Figure 1.)

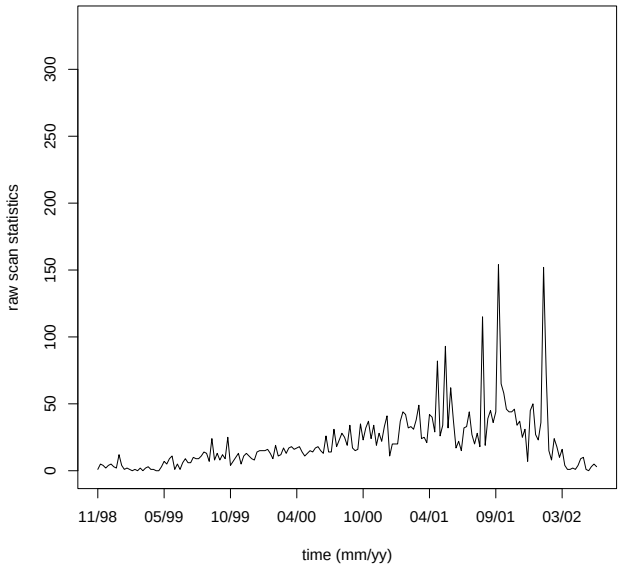


Figure 10: Time series of scan statistic $M_{1,t}$ for weekly Enron email digraphs during a period of 189 weeks from 1998–2002. (See also Figure 1.)

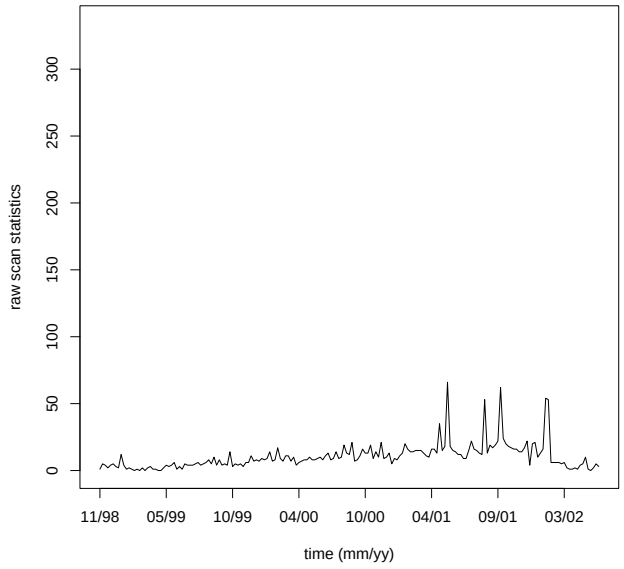


Figure 9: Time series of scan statistic $M_{0,t}$ (max degree) for weekly Enron email digraphs during a period of 189 weeks from 1998–2002. (See also Figure 1.)

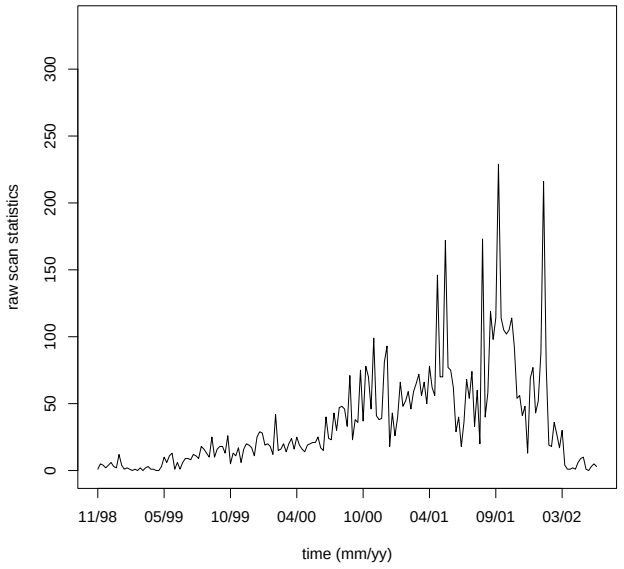


Figure 11: Time series of scan statistic $M_{2,t}$ for weekly Enron email digraphs during a period of 189 weeks from 1998–2002. (See also Figure 1.)

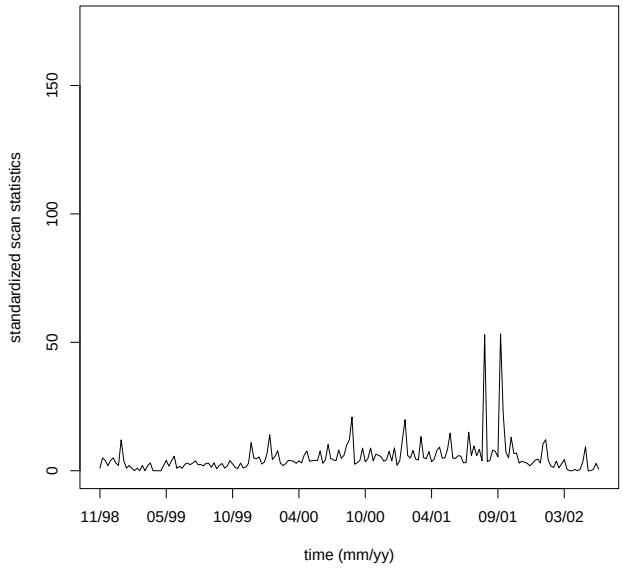


Figure 12: Time series of standardized scan statistic $\widetilde{M}_{0,t}$ for weekly Enron email digraphs during a period of 189 weeks from 1998–2002. (See also Figure 2.)

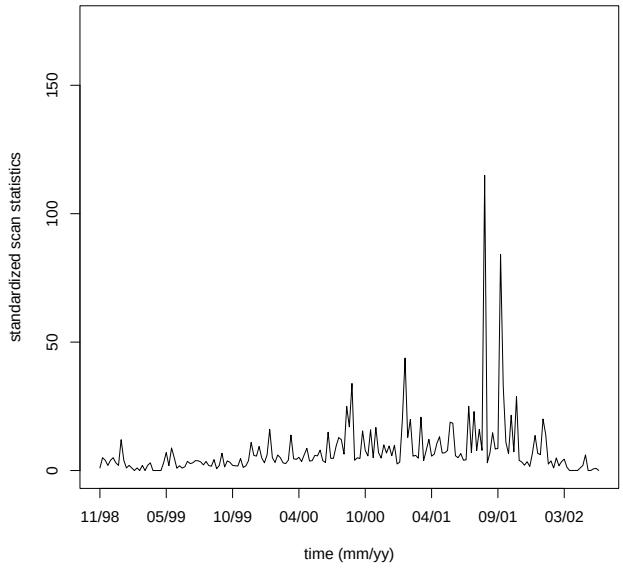


Figure 13: Time series of standardized scan statistic $\widetilde{M}_{1,t}$ for weekly Enron email digraphs during a period of 189 weeks from 1998–2002. (See also Figure 2.)

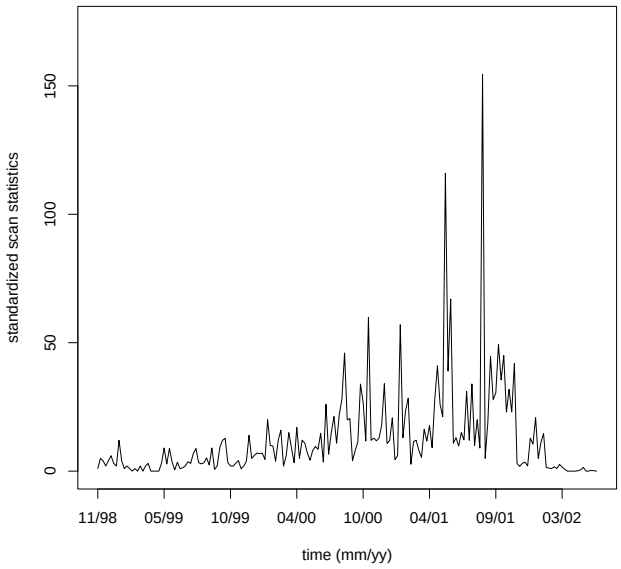


Figure 14: Time series of standardized scan statistic $\widetilde{M}_{2,t}$ for weekly Enron email digraphs during a period of 189 weeks from 1998–2002. (See also Figure 2.)

The Author-Recipient-Topic Model for Topic and Role Discovery in Social Networks, with Application to Enron and Academic Email

Andrew McCallum, Andrés Corrada-Emmanuel, Xuerui Wang
Department of Computer Science
University of Massachusetts Amherst
Amherst, MA 01003, USA
{mccallum,corrada,xuerui}@cs.umass.edu

Abstract

Previous work in social network analysis (SNA) typically models the existence of links from one entity to another, but not the language content or topics on those links. We present the Author-Recipient-Topic (ART) model for social network analysis, which learns topic distributions based on the the direction-sensitive messages sent between entities. The model builds on Latent Dirichlet Allocation and the Author-Topic (AT) model, adding the key attribute that distribution over topics is conditioned distinctly on both the sender and recipient—steering the discovery of topics according to the relationships between people. We give results on both the Enron email corpus and a researcher's email archive, providing evidence not only that clearly relevant topics are discovered, but that the ART model better predicts people's roles.

Keywords: Social network analysis, language modeling, topic discovery, graphical models, Gibbs sampling.

1 Introduction

Social network analysis (SNA) is the study of mathematical models for interactions among people, organizations and groups. With the recent availability of large datasets of human interactions (Shetty & Adibi, 2004; Wu et al., 2003), the popularity of services like Friendster.com and LinkedIn.com, and the salience of the connections among the 9/11 hijackers, there has been growing interest in social network analysis.

Historically, research in the field has been led by social scientists and physicists (Lorrain & White, 1971; Albert & Barabási, 2002; Watts, 2003; Wasserman & Faust, 1994; Carley, 1991), and previous work has emphasized binary interaction data, sometimes with directed edges, sometimes with weights on the edges. There has not, however, yet been significant work by researchers with backgrounds in statistical natural language processing, nor analysis that captures the richness of the *language contents* of the interactions—the words, the topics, and other high-dimensional specifics of the messages between people.[1]

Using pure network connectivity properties, SNA often aims to discover various categories of nodes in a network. For example, in addition to determining that a node-degree distribution is heavy-tailed, we can also find those particular nodes with an inordinately high number of connections, or with connections to a particularly well-connected subset of the network. Furthermore, using these properties we can assign "roles" to certain nodes, *e.g.* (Lorrain & White, 1971; Wolfe & Jensen, 2003). However, it is clear that network properties are not enough to discover all the roles in a social network. Consider email messages in a corporate setting, and imagine a situation where a tightly knit group of users trade email messages with each other in a roughly symmetric fashion. Thus, at the network level they appear to fulfill the same role. But perhaps, one of the users is in fact a manager for the whole group—a role that becomes obvious only when one accounts for the language content of the email messages.

Outside of the social network analysis literature, there has been a stream of new research in machine learning and natural language models for clustering words in order to discover the few underlying topics that are combined to form documents in a corpus. Latent Dirichlet Allocation (Blei et al., 2003) can be run on thousands or millions of words of text data to automatically and robustly discover multinomial word distributions of these topics. Hierarchical Dirichlet Processes (Teh et al., 2004) can determine an appropriate number of topics for a corpus. The Author-Topic Model (Steyvers et al., 2004; Rosen-Zvi et al., 2004) learns topics conditioned on the mixture of authors that composed a document. However, none of these models are appropriate for social network analysis, in which we aim to capture the directed interactions and relationships between people.

[1] A recent paper by Diesner and Carley (2004b) describes a method for analyzing text to produce a social network. We discuss this and surrounding work in the Related Work section below.

The paper presents the *Author-Recipient-Topic* (ART) model, a directed graphical model of words in a message generated given their author and a set of recipients. The model is similar to the Author-Topic (AT) model, but with the crucial enhancement that it conditions the per-message topic distribution jointly on both the author and individual recipients, rather than on individual authors. Thus the discovery of topics in the ART model is influenced by the social structure in which messages are sent and received. Each topic consists of a multinomial distribution over words. Each author-recipient pair has a distribution over topics. We can also easily calculate marginal distributions over topics conditioned solely on an author, or solely on a recipient, in order to find the topics on which each person is most likely to send or receive.

Most importantly, we can also effectively use these person-conditioned topic distributions to measure similarity between people, and thus discover people's roles by clustering using this similarity.[2] For example, people who receive messages containing requests for photocopying, travel bookings, and meeting room arrangements can all be said to have the role "administrative assistant," and can be discovered as such because in the ART model they will all have these topics with high probability in their receiving distribution. Note that we can discover that two people have similar roles even if in the graph they are connected to very different sets of people.

We demonstrate this model on the Enron email corpus comprising 147 people and 24k messages, and also on 9 months of incoming and outgoing mail of the first author, comprising 825 people and 23k messages. We show not only that ART discovers extremely salient topics, but also give evidence that ART predicts people's roles better than AT. Furthermore we show that the similarity matrix produced by AT is drastically different than the SNA matrix, but ART's is similar, while also having some interesting differences.

We also describe an extension of the ART model that explicitly captures *roles* of people, by generating role associations for the authors and recipients of a message, and conditioning the topic distributions on those role assignments. The model, which we term Role-Author-Recipient-Topic (RART), naturally represents that one person can have more than one role. We present three possible RART variants, and describe preliminary experiments with one of these variants.

[2]The clustering may either external to the model by simple greedy-agglomerative clustering, or internal to the model by introducing latent variables for the sender's and recipient's roles, as described in the Role-Author-Recipient-Topic (RART) model toward the end of this paper.

2 Author-Recipient-Topic Models

Before describing the Author-Recipient-Topic model, we first describe three related models. Latent Dirichlet Allocation (LDA) is a Bayesian network that generates a document using a mixture of topics (Blei et al., 2003). In its generative process, for each document d, a multinomial distribution θ over topics is randomly sampled from a Dirichlet with parameter α, and then to generate each word, a topic z is chosen from this topic distribution, and a word, w, is generated by randomly sampling from a topic-specific multinomial distribution ϕ_z. The robustness of the model is greatly enhanced by integrated out uncertainty about the per-document topic distribution θ.

The Author model (also termed a Multi-label Mixture Model) (McCallum, 1999), is a Bayesian network that simultaneously models document content and its authors' interests with a one-to-one correspondence between topics and authors. The model was originally applied to multi-label document classification, with categories acting as authors. In its generative process, for each document a set of authors $\mathbf{a}_d$ is observed. To generate each word, an author, z, is sampled uniformly from the set, and then a word, w, is generated by sampling from an author-specific multinomial distribution ϕ_z.

The Author-Topic (AT) model is a similar Bayesian network, in which each authors' interests are modeled with a *mixture* of topics (Steyvers et al., 2004; Rosen-Zvi et al., 2004). In its generative process for each document, a set of authors, $\mathbf{a}_d$, is observed. To generate each word, an author x is chosen at uniform from this set, then a topic z is selected from a topic distribution θ_x that is specific to the author, and then a word w is generated by sampling from a topic-specific multinomial distribution ϕ_z.

However, as described previously, neither of these models are suitable for modeling message data.

An email message has one sender and in general more than one recipients. We could treat both the sender and the recipients as "authors" of the message, and then employ the AT model, but it does not distinguish the author and the recipients of the message. This is undesirable in many real-world situations. A manager may send email to a secretary and vice versa, but the nature of the requests and language used may be quite different. Even more dramatically, consider the large quantity of junk email that we receive; modeling the topics of these messages as undistinguished from the topics we write about as authors would be extremely confounding and undesirable since they do not reflect our expertise or roles.

Alternatively we could still employ the AT model by ignoring the recipient information of email and treating each email document as if it only has one author.

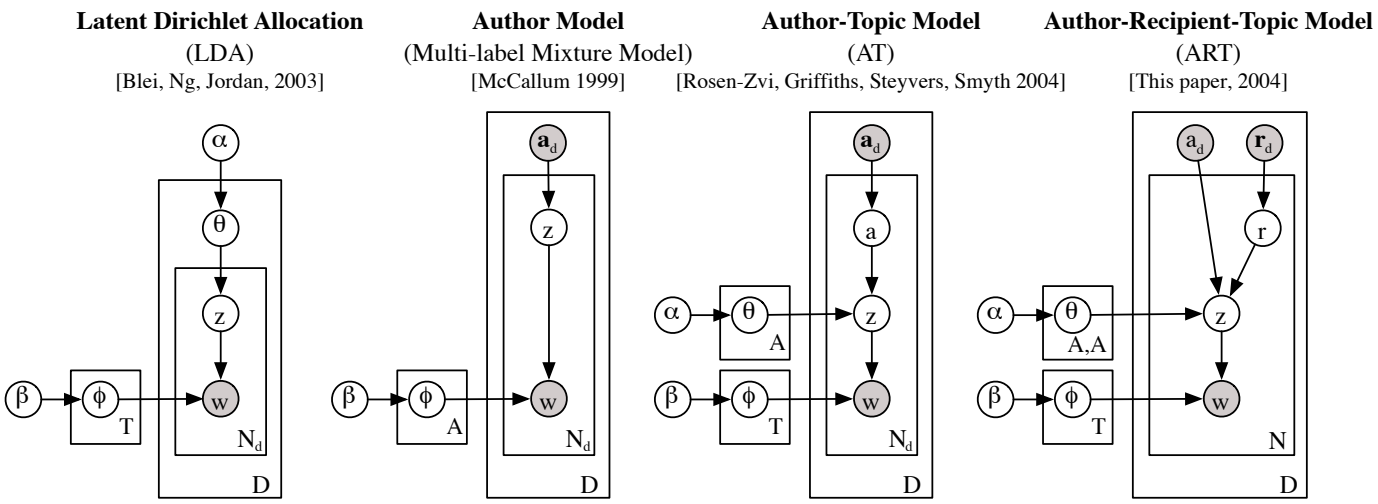


Figure 1: Three related models, and the Author-Recipient-Topic model. In all models, each observed word, w, is generated from a multinomial word distribution, ϕ_z, specific to a particular topic, z, however topics are selected differently in each of the models. In LDA, the topic is sampled from a per-document topic distribution, θ, which in turn is sampled from a Dirichlet over topics. In the Author Model, there is one topic associated with each author (or category), and authors are sampled uniformly. In the Author-Topic model, there is a separate topic-distribution, θ, for each author, and the selection of topic-distribution is determined by uniformly sampling an author from the observed list of the document's authors. In the Author-Recipient-Topic model, there is a separate topic-distribution for each author-recipient pair, and the selection of topic-distribution is determined from the observed author, and by uniformly sampling from the set of recipients for the document.

However, in this case (which is similar to the LDA model) we are lose all information about the recipients, and the connections between people implied by sender-recipient relationships.

Thus, we propose an Author-Recipient-Topic (ART) model for message data. The ART model captures topics and the directed social network of senders and receivers by conditioning the multinomial distribution over topics distinctly on both the author and one recipient of a message. Unlike the AT, the ART model takes into consideration both author and recipients distinctly, in addition to modeling the email content as a mixture of topics.

The ART model is a Bayesian network that simultaneously models message content, as well as the directed social network in which the messages are sent. In its generative process for each message, an author, a_d, and a set of recipients, $\mathbf{r}_d$, are observed. To generate each word, a recipient, x, is chosen at uniform from $\mathbf{r}_d$, and then a topic z is chosen from a multinomial topic distribution $\theta_{a_d,x}$, where the distribution is specific to the author-recipient pair (a_d, x). (This distribution over topics could also be smoothed against a distribution conditioned on the author only, although we did not find that to be necessary in our experiments.) Finally, the word w is generated by sampling from a topic-specific multinomial distribution ϕ_z. The result is that the discovery of topics is guided by the social network in which the collection of message text was generated.

The Bayesian network for all four models is shown in Figure 1.

In the ART model, for a particular message d, given the hyperparameters of the Dirichlet distributions α and β (assumed fixed in this paper), the author a_d, and the set of recipients $\mathbf{r}_d$, the joint distribution of an author mixture θ, a topic mixture ϕ, a set of N_d recipients $\mathbf{x}_d$, a set of N_d topics $\mathbf{z}_d$ and a set of N_d words $\mathbf{w}_d$ is given by:

$$p(\theta, \phi, \mathbf{x}_d, \mathbf{z}_d, \mathbf{w}_d | \alpha, \beta, a_d, \mathbf{r}_d) =$$
$$p(\theta|\alpha)p(\phi|\beta) \prod_{n=1}^{N_d} p(x_{dn}|\mathbf{r}_d)p(z_{dn}|\theta_{a_d,x_{dn}})p(w_{dn}|\phi_{z_{dn}})$$

Integrating over θ and ϕ, and summing over $\mathbf{x}_d$ and $\mathbf{z}_d$, we get the marginal distribution of a document:

$$p(\mathbf{w}_d|\alpha, \beta, a_d, \mathbf{r}_d) =$$
$$\iint p(\theta|\alpha)p(\phi|\beta)$$
$$\prod_{n=1}^{N_d} \sum_{x_{dn}} \sum_{z_{dn}} p(x_{dn}|\mathbf{r}_d)p(z_{dn}|\theta_{a_d,x_{dn}})p(w_{dn}|\phi_{z_{dn}}) \mathrm{d}\phi \mathrm{d}\theta$$

Finally, we take the product of the marginal probabilities of single documents, and the probability of a corpus is:

$$p(\mathbf{D}|\alpha,\beta,\mathbf{a},\mathbf{r}) = \prod_{d=1}^{D} p(\mathbf{w}_d|\alpha,\beta,a_d,\mathbf{r}_d)$$

2.1 Monte Carlo Gibbs sampling Inference on models in the LDA family cannot be performed exactly. Three standard approximations have been used to obtain practical results: variational methods (Blei et al., 2003), Gibbs sampling (Griffiths & Steyvers, 2004; Steyvers et al., 2004; Rosen-Zvi et al., 2004), and expectation propagation (Griffiths & Steyvers, 2004; Minka & Lafferty, 2002). We chose Gibbs sampling for its ease of implementation.

To carry out the Gibbs sampling we need to derive a formula for $P(z_i, x_i|\mathbf{z}_{-i}, \mathbf{x}_{-i})$, the conditional distribution of a topic and recipient for the i_w word given all other words topic and recipient assignments, $\mathbf{z}_{-i}$ and $\mathbf{x}_{-i}$. To understand why, let us try to calculate $P(\mathbf{z}, \mathbf{x} \,|\, \mathbf{w})$, the posterior distribution of topic and recipient assignments given the words in the corpus.

We begin by calculating $P(\mathbf{w} \,|\, \mathbf{z}, \mathbf{x})$. Let T be the number of topics, W be the vocabulary size, and V be the number of all word tokens. Using $P(\mathbf{w} \,|\, \mathbf{z}, \mathbf{x}, \Phi)$, we can integrate out the unknown Φ distributions to obtain:

$$P(\mathbf{w} \,|\, \mathbf{z}, \mathbf{x}, \Phi) = \prod_{i_w=1}^{W} \phi_{z_{i_w}}(w_{i_w})$$

Rearranging the product over the W word tokens present in the corpus to collect words that are assigned to the same topic, we obtain,

$$P(\mathbf{w} \,|\, \mathbf{z}, \mathbf{x}, \Phi) = \prod_{z=1}^{T} \prod_{v=1}^{V} \phi_z^{n_z^{w_v}},$$

where $n_z^{w_v}$ is the number of times that a vocabulary word, w_v, was assigned to a topic. And finally, we integrate out the ϕ distributions by using the Dirichlet distribution,

$$\begin{aligned} &P(\mathbf{w} \,|\, \mathbf{z}, \mathbf{x}) \\ &= \int \prod_{z=1}^{T} \left(\frac{\Gamma(\sum_{v=1}^{V} \beta_v)}{\prod_{v=1}^{V} \Gamma(\beta_v)} \left(\prod_{v=1}^{V} \phi_z^{n_z^{w_v}+\beta_v-1}(w_v) d\phi_z(w_v) \right) \right) \\ &= \prod_{z=1}^{T} \left(\frac{\Gamma(\sum_{v=1}^{V} \beta_v)}{\prod_{v=1}^{V} \Gamma(\beta_v)} \left(\frac{\prod_{v=1}^{V} \Gamma(n_z^{w_v}+\beta_v)}{\Gamma(\sum_{v=1}^{V} \beta_v + \sum_{v=1}^{V} n_z^{w_v})} \right) \right) \end{aligned}$$

Similarly, we can calculate $P(\mathbf{z}, \mathbf{x})$ using a procedure analogous to that used for $P(\mathbf{w} \,|\, \mathbf{z}, \mathbf{x})$. We collect terms from vocabulary words assigned to the same topic and author-recipient pair and integrate out the Θ distributions corresponding to all the different author-recipient pairs, P:

$$P(\mathbf{z}, \mathbf{x}) = \left(\prod_{i_w=1}^{W} \frac{1}{n_R(d_{i_w})} \right) \prod_{p=1}^{P} \left(\frac{\Gamma(\sum_z \alpha_z)}{\prod_{z=1}^{T} \Gamma(\alpha_z)} \frac{\prod_z \Gamma(n_p^z + \alpha_z)}{\Gamma(\sum_z \alpha_z + \sum_z n_p^z)} \right),$$

where $n_R(d_{i_w})$ is the number of recipients corresponding to a word in a given email.

Putting together our equations for $P(\mathbf{w} \,|\, \mathbf{z}, \mathbf{x})$ and $P(\mathbf{z}, \mathbf{x})$ we can obtain an expression for $P(\mathbf{w}, \mathbf{z}, \mathbf{x})$. This allows us to write an expression for the posterior distribution of $\mathbf{z}$ and $\mathbf{x}$ given the corpus,

$$P(\mathbf{z}, \mathbf{x} \,|\, \mathbf{w}) = \frac{P(\mathbf{w}, \mathbf{z}, \mathbf{x})}{\sum_{\mathbf{z},\mathbf{x}} P(\mathbf{w}, \mathbf{z}, \mathbf{x})}$$

However, we cannot calculate the denominator directly.

Gibbs sampling gets around this intractability by using the conditional distribution $P(z_i, x_i, w_i \,|\, \mathbf{z}_{-i}, \mathbf{x}_{-i}, \mathbf{w}_{-i})$ to run a Markov chain Monte Carlo calculation. We can calculate this conditional as,

$$\begin{aligned} &P(z_i, x_i, w_i \,|\, \mathbf{z}_{-i}, \mathbf{x}_{-i}, \mathbf{w}_{-i}) \\ &\quad = \frac{P(\mathbf{z}, \mathbf{x}, \mathbf{w})}{P(\mathbf{z}_{-i}, \mathbf{x}_{-i}, \mathbf{w}_{-i})} \\ &\quad = \frac{1}{n_R} \frac{\frac{\Gamma(n_p^t+\alpha_t)}{\Gamma(\sum_z n_p^z+\sum_z \alpha_z)} \frac{\Gamma(n_t^{w_v}+\beta_v)}{\Gamma(\sum_v n_t^{w_v}+\sum_v \beta_v)}}{\frac{\Gamma(n_p^t-1+\alpha_t)}{\Gamma(\sum_z n_p^z-1+\sum_z \alpha_z)} \frac{\Gamma(n_t^{w_v}-1+\beta_v)}{\Gamma(\sum_v n_z^{w_v}-1+\sum_v \beta_v)}} \\ &\quad = \frac{1}{n_R} \frac{n_{p,-i}^t + \alpha_t}{\sum_z n_{p,-i}^z + \sum_z \alpha_z} \frac{n_{t,-i}^{w_v} + \beta_v}{\sum_v n_{t,-i} + \sum_v \beta_v}, \end{aligned}$$

where the recipient, r, is part of the author-recipient pair, p, the $-i$ subscript is used to denote that the counts are taken by excluding the assignment of word i itself, and n_R is the number of recipients for the email to which word i belongs.

Further manipulation can turn the above equation into update equations for the topic and recipient of each corpus token, $P(z_i \,|\, \mathbf{z}_{-i}, \mathbf{x}, \mathbf{w})$ and $P(x_i \,|\, \mathbf{z}, \mathbf{x}_{-i}, \mathbf{w})$ suitable for random or systematic scan updates:

$$\begin{aligned} P(z_i \,|\, \mathbf{z}_{-i}, \mathbf{x}, \mathbf{w}) &\propto \frac{n_{z_i}^{w_v} + \beta_v}{\sum_v n_{z_i}^{w_v} + \beta_v} \frac{n_{x_i}^{z_i} + \alpha_{z_i}}{\sum_{z'} n_{x_i}^{z'} + \alpha_{z'}} \\ P(x_i \,|\, \mathbf{z}, \mathbf{x}_{-i}, \mathbf{w}) &\propto \frac{n_{x_i}^{z_i} + \alpha_{z_i}}{\sum_{z'} n_{x_i}^{z'} + \alpha_{z'}} \end{aligned}$$

3 Related Work

The use of social networks to discover "roles" for the people (or nodes) in the network goes back over three decades to the work of Lorrain and White (1971). It is based on the hypothesis that nodes on a network that relate to other nodes in "equivalent" ways must have the

same role. This equivalence was given a probabilistic interpretation by Holland et al. (1983): nodes assigned to a class/role are stochastically equivalent if the probabilities of the relationships with all other nodes are the same for nodes in the same class/role.

The limitation of a single class/role label for each node in the network was relaxed in recent work by Wolfe and Jensen (2003). They consider a model that assigns multiple role labels to a given node in the network. One advantage of multiple labels is that in this factored model, fewer parameters are required to be estimated than in a non-factored, single label obliged to represent more values. They find that, two labels with three values (giving $3^2 = 9$ possible labelings for each node) is a better estimator for synthetic data produced by a two-label process than a model using one label with nine possible values. This is, of course, the advantage of *mixture models*, such as LDA and the ART model presented here.

There has been some work that processes text to perform network analysis. *Network Text Analysis* (NTA), *e.g.* Diesner et al. (2003), takes text as input, and in a semi-automated fashion produces a network in which the nodes are *words* or concepts. First the input text is preprocessed—removing uninteresting words, manually conjoining words into phrases where appropriate, and manually making word substitutions to collapse similar concepts. Then a sliding window of fixed width (say 8) is moved across the remaining sequence of words and phrases, and words or phrases that co-occur in the window are connected by a binary edge, thus forming a graph of words called a *cognitive map*. This approach as been applied to the text of student answers to questionnaires (Carley, 1997). By measuring the degree of overlap between the networks produced from different students answers, one may attempt to discover the degree of similarity between the students' conceptualizations of the subject matter. AutoMap1.2 (Diesner & Carley, 2004a) is a later version is more automated and employs named entity recognition as part of its preprocessing. *MetaMatrix Analysis* (Krackhardt & Carley, 1998; Carley, 2003) uses word-lists or thesauri to assign categories to the output of AutoMap; the categories include agent, knowledge, resource, task-event, and organization. AutoMap has also been applied to LexisNexus text about people from a certain region to discover networks of people in certain relations to MetaMatrix-categorized concepts. Note that the NTA and AutoMap methods are sensitive to word proximity in the text stream, and the window size.

In contrast to AutoMap, the ART model described in this paper uses statistics from text messages sent between entities, rather than word proximity in a sliding window of text. In other words, NTA methods build a network of words; the ART model consumes the network given by a corpus of message data, its authors and recipients. ART also takes a more automated approach—although it certainly would be interesting to consider opportunities for human input into the process. Since ART is based on robustly fitting a formal probabilistic model to statistics from large-scale data, and since it employs Bayesian methods to integrate out uncertainty about hidden variables, one should expect its output to be more robust to inherent noise in the data.

The study of email social networks has been hampered by the previous unavailability of a public corpus. The research that has been published has used email to-from logs. Logs are easier to obtain and are less intrusive on user's privacy. This means that previous research has focused on the topological structure of email networks, and the dynamics of the email traffic between users. Wu et al. (2003) looked at how information flowed in an email network of users in research labs (mostly from HP Labs). They conclude that epidemic models of information flow do not work for email networks and thus identifying hubs in the network may not guarantee that information originating at a node reaches a large fraction of the network. This finding serves as an example that network properties are not sufficient to optimize flow on an email network. Adamic and Adar (2004) studied the efficiency of "local information" search strategies on social networks. They find that in the case of an email network at HP Labs, a greedy search strategy works efficiently as predicted by Kleinberg (2000) and Watts et al. (2002).

All these approaches, however, limit themselves to the use of network topology to discover roles. The ART model complements these approaches by using the content of the "traffic" between nodes to create language models that can bring out differences invisible at the network level.

As discussed above, the ART model is a direct offspring of Latent Dirichlet Allocation (Blei et al., 2003), the Multi-label Mixture Model (McCallum, 1999), and the Author-Topic Model (Steyvers et al., 2004; Rosen-Zvi et al., 2004), with the distinction that ART is specifically designed to capture language used in a directed network of correspondents.

4 Experimental Results

We present results with the Enron email corpus and the personal email of one of the authors of this paper (McCallum). The Enron email corpus, is a large body of email messages subpoenaed as part of the investigation by the Federal Energy Regulatory Commission (FERC), and then placed in the public record. The original dataset contains 517,431 messages, however MD5

Topic 5 "Legal Contracts"		Topic 17 "Document Review"		Topic 27 "Time Scheduling"		Topic 45 "Sports Pool"	
section	0.0299	attached	0.0742	day	0.0419	game	0.0170
party	0.0265	agreement	0.0493	friday	0.0418	draft	0.0156
language	0.0226	review	0.0340	morning	0.0369	week	0.0135
contract	0.0203	questions	0.0257	monday	0.0282	team	0.0135
date	0.0155	draft	0.0245	office	0.0282	eric	0.0130
enron	0.0151	letter	0.0239	wednesday	0.0267	make	0.0125
parties	0.0149	comments	0.0207	tuesday	0.0261	free	0.0107
notice	0.0126	copy	0.0165	time	0.0218	year	0.0106
days	0.0112	revised	0.0161	good	0.0214	pick	0.0097
include	0.0111	document	0.0156	thursday	0.0191	phillip	0.0095
M.Hain	0.0549	G.Nemec	0.0737	J.Dasovich	0.0340	E.Bass	0.3050
J.Steffes		B.Tycholiz		R.Shapiro		M.Lenhart	
J.Dasovich	0.0377	G.Nemec	0.0551	J.Dasovich	0.0289	E.Bass	0.0780
R.Shapiro		M.Whitt		J.Steffes		P.Love	
D.Hyvl	0.0362	B.Tycholiz	0.0325	C.Clair	0.0175	M.Motley	0.0522
K.Ward		G.Nemec		M.Taylor		M.Grigsby	

Topic 34 "Operations"		Topic 37 "Power Market"		Topic 41 "Government Relations"		Topic 42 "Wireless"	
operations	0.0321	market	0.0567	state	0.0404	blackberry	0.0726
team	0.0234	power	0.0563	california	0.0367	net	0.0557
office	0.0173	price	0.0280	power	0.0337	www	0.0409
list	0.0144	system	0.0206	energy	0.0239	website	0.0375
bob	0.0129	prices	0.0182	electricity	0.0203	report	0.0373
open	0.0126	high	0.0124	davis	0.0183	wireless	0.0364
meeting	0.0107	based	0.0120	utilities	0.0158	handheld	0.0362
gas	0.0107	buy	0.0117	commission	0.0136	stan	0.0282
business	0.0106	customers	0.0110	governor	0.0132	fyi	0.0271
houston	0.0099	costs	0.0106	prices	0.0089	named	0.0260
S.Beck	0.2158	J.Dasovich	0.1231	J.Dasovich	0.3338	R.Haylett	0.1432
L.Kitchen		J.Steffes		R.Shapiro		T.Geaccone	
S.Beck	0.0826	J.Dasovich	0.1133	J.Dasovich	0.2440	T.Geaccone	0.0737
J.Lavorato		R.Shapiro		J.Steffes		R.Haylett	
S.Beck	0.0530	M.Taylor	0.0218	J.Dasovich	0.1394	R.Haylett	0.0420
S.White		E.Sager		R.Sanders		D.Fossum	

Table 1: An illustration of several topics from a 50-topic run for the Enron Email Dataset. Each topic is shown with the top 10 words and their corresponding conditional probabilities. The quoted titles are our own summary for the topics. Below are prominent author-recipient pairs for each topic. For example, Mary Hain was an in-house lawyer at Enron; Eric Bass was the coordinator of a fantasy basketball league within Enron. In the "Operations" topic it is satisfying to see Beck, who was the Chief Operating Officer at Enron; Kitchen was President of Enron Online; and Lavorato was CEO of Enron America. In the "Government Relations" topic, we see Dasovich, who was a Government Relation Executive, Shapiro who was Vice President of Regulatory Affairs, Steffes, who was Vice President of Government Affairs, and Sanders, who was Vice President of WholeSale Services. In "Wireless" we see that Haylett, who was Chief Financial Officer and Treasurer, was an avid user of the Blackberry brand wireless, portable email system.

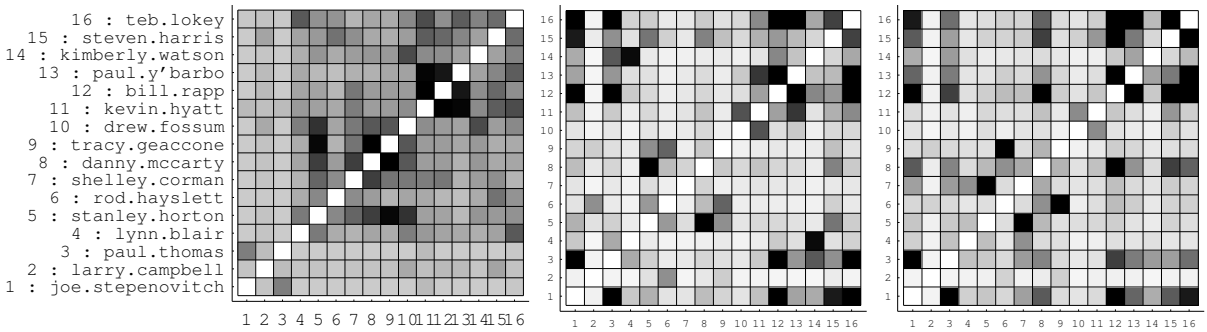


Figure 2: **Left:** SNA Inverse JS Network. **Middle:** ART Inverse JS Network. **Right:** AT Inverse JS Network. Darker shades indicate higher similarity.

hashes on contents, authors and dates show only 250,484 of these to be unique.

Although the Enron Email Dataset contains the email folders of 150 people, two people appear twice with different usernames, and we remove one person who only sent automated calendar reminders, resulting in 147 people for our experiments. We hand-corrected variants of the email addresses for these 147 users to capture the connectivity of as much of these users' email as possible. The total number of email messages traded among these users is 23,488. We did not model email messages that were not received by at least one of the 147.

In order to capture only the new text entered by the author of a message, it is necessary to remove "quoted original messages" in replies. We eliminate this extraneous text by a simple heuristic: all text in a message below a "forwarded message" line or timestamp is removed. This heuristic certainly incorrectly looses words that are interspersed with quoted email text. Words are formed as sequences of alphabetic characters. To remove sensitivity to capitalization, all text is downcased.

Our second dataset consists of the personal email sent and received by McCallum between January and October 2004. It consists of 23,488 unique messages written by 825 authors. In typical power-law behavior, most of these authors wrote only a few messages, while 128 wrote ten or more emails. After applying the same text normalization filter (lowercasing, removal of quoted email text, etc.) that was used for the Enron data, we obtained a text corpus containing 457,057 word tokens, and a vocabulary of 22,901 unique words.

To simplify the Gibbs formulae, we keep the hyperparameters of the Dirichlet distributions symmetric, that is, here setting all dimensions of α to 50 and β to 0.01.

4.1 Topics and Prominent Relations from ART models Table 1 shows the highest probability words from eight topics in an ART model trained on the 147 users with 50 topics. (The quoted titles are our own interpretation of a summary for the topics.) The clarity and specificity of these topics are typical of the topics discovered by the model. For example, Topic 17 comes from message discussing review and comments on documents; Topic 27 comes from messages negotiating meeting times.

Beneath the word distribution for each topic are the three author-recipient pairs with highest probability of discussing that topic—each pair separated by a horizontal line, with the author above the recipient. For example, Mary Hain, the top author of messages in the "Legal Contracts" topic, was an in-house lawyer at Enron. By inspection of other messages, Eric Bass seems to have been the coordinator for a fantasy basketball league among Enron employees.

4.2 Stochastic Blockstructures and Roles The stochastic equivalence hypothesis from SNA states that nodes in a network that behave stochastically equivalently must have similar roles. In the case of an email network consisting of message counts, the natural way to measure equivalence is to examine the probability that a node communicated with other nodes. If two nodes have similar probability distribution over their communication partners, we should consider them role-equivalent. Lacking a true distance measure between probability distributions, we can use some symmetric measure, such as the Jensen-Shannon (JS) divergence, to obtain a symmetric matrix relating the nodes in the network. Since we want to consider nodes/users that have a small JS divergence as equivalent, we can use the inverse of the divergence to construct a symmetric matrix in which larger numbers indicate higher similarity between users.

Standard recursive graph-cutting algorithms on this matrix can be used to cluster users, rearranging the rows/columns to form approximately block-diagonal

Topic 5 "Grant Proposals"		Topic 31 "Meeting Setup"		Topic 38 "ML Models"		Topic 41 "Friendly Discourse"	
proposal	0.0397	today	0.0512	model	0.0479	great	0.0516
data	0.0310	tomorrow	0.0454	models	0.0444	good	0.0393
budget	0.0289	time	0.0413	inference	0.0191	don	0.0223
work	0.0245	ll	0.0391	conditional	0.0181	sounds	0.0219
year	0.0238	meeting	0.0339	methods	0.0144	work	0.0196
glenn	0.0225	week	0.0255	number	0.0136	wishes	0.0182
nsf	0.0209	talk	0.0246	sequence	0.0126	talk	0.0175
project	0.0188	meet	0.0233	learning	0.0126	interesting	0.0168
sets	0.0157	morning	0.0228	graphical	0.0121	time	0.0162
support	0.0156	monday	0.0208	random	0.0121	hear	0.0132
smyth mccallum	0.1290	ronb mccallum	0.0339	casutton mccallum	0.0498	mccallum culotta	0.0558
mccallum stowell	0.0746	wellner mccallum	0.0314	icml04-webadmin icml04-chairs	0.0366	mccallum casutton	0.0530
mccallum lafferty	0.0739	casutton mccallum	0.0217	mccallum casutton	0.0343	mccallum ronb	0.0274
mccallum smyth	0.0532	mccallum casutton	0.0200	nips04workflow mccallum	0.0322	mccallum saunders	0.0255
pereira lafferty	0.0339	mccallum wellner	0.0200	weinman mccallum	0.0250	mccallum pereira	0.0181

Table 2: The four topics most prominent in McCallum's email exchange with Padhraic Smyth, from a 50-topic run of ART on 10 months of McCallum's email. The topics provide an extremely salient summary of McCallum and Smyth's relationship during this time period: they wrote a grant proposal together; they set up many meetings; they discussed machine learning models; they were friendly with each other. Each topic is shown with the 10 highest-probability words and their corresponding conditional probabilities. The quoted titles are our own summary for the topics. Below are prominent author-recipient pairs for each topic. The people other than smyth also appear in very sensible associations: stowell is McCallum's proposal budget administrator; McCallum also wrote a proposal with John Lafferty and Fernando Pereira; McCallum also sets up meetings, discusses machine learning and has friendly discourse with his graduate student advisees: ronb, wellner, casutton, and culotta; he does not, however, discuss the details of proposal-writing with them.

structures. This is the familiar process of 'blockstructuring' used in SNA. We perform such an analysis on two datasets: a small subset of the Enron users consisting mostly of people associated with the Transwestern Pipeline Division within Enron, and the entirety of McCallum's email.

We begin with the Enron TransWestern Pipeline Division. Our analysis here employed a "closed-universe" assumption—only those messages traded among authors in the dataset were used.

The traditional SNA similarity measure (in this case JS divergence of distributions on recipients from each person) is shown in the left matrix in Figure 2. Darker shading indicates that two users are considered more similar. A related matrix resulting from our ART model (JS divergence of recipient-marginalized topic distributions for each email author) appears in the middle of the Figure. Finally, the results of the same analysis using topics from the AT model rather than our ART model can be seen on the right. The three matrices are similar, but have interesting differences.

Consider Enron employee Geaccone (user 9 in all the matrices in Figure 2). According to the traditional SNA role measurement, Geaccone and McCarty (user 8) have very similar roles, however, both the AT and ART models indicate no special similarity. Inspection of the email messages for both users reveals that Geaconne was an Executive Assistant, while McCarty was a Vice-President—rather different roles—and, thus output of ART and AT is more appropriate. We can interpret these results as follows: SNA analysis shows that they wrote email to similar sets of people, but the ART analysis illustrates that they used very different language when they wrote to these people.

Comparing ART against AT, both models provide similar role distance for Geaccone versus McCarty, but ART and AT show their differences elsewhere. For example, AT indicates a very strong role similarity between Geaconne and Hayslett (user 6), who was her boss (and CFO & Vice President in the Division);

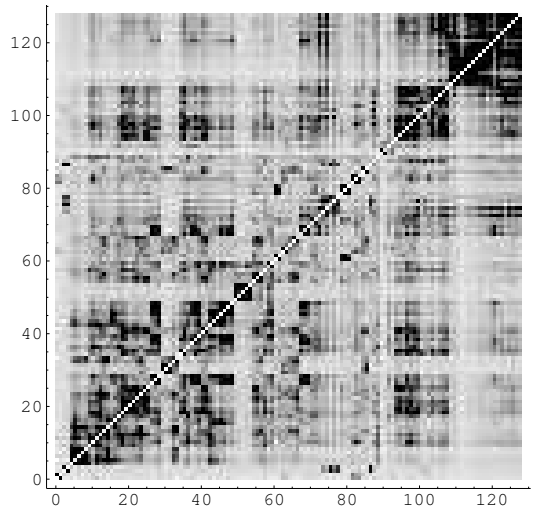


Figure 3: SNA Inverse JS Network for a 10 topic run on McCallum Email Data. Darker shades indicate higher similarity. Graph partitioning was calculated with the 128 authors that had ten or more emails in McCallum's Email Data. The block from 0 to 30 are people in and related to McCallum's research group at UMass. The block from 30 to 50 includes other researchers around the world.

Pairs considered most alike by ART	
User Pair	*Description*
editor reviews	Both journal review management
mike mikem	Same person! (manual coref error)
aepshtey smucker	Both students in McCallum's class
coe laurie	Both UMass admin assistants
mcollins tom.mitchell	Both ML researchers on SRI project
mcollins gervasio	Both ML researchers on SRI project
davitz freeman	Both ML researchers on SRI project
mahadeva pal	Both ML researchers, discussing hiring
kate laurie	Both UMass admin assistants
ang joshuago	Both on org committee for a conference

Pairs considered most alike by SNA	
User Pair	*Description*
aepshtey rasmith	Both students in McCallum's class
donna editor	Spouse is unrelated to journal editor
donna krishna	Spouse is unrelated to conference organizer
donna ramshaw	Spouse is unrelated to researcher at BBN
donna reviews	Spouse is unrelated to journal editor
donna stromsten	Spouse is unrelated to visiting researcher
donna yugu	Spouse is unrelated grad student
aepshtey smucker	Both students in McCallum's class
rasmith smucker	Both students in McCallum's class
editor elm	Journal editor and its Production Editor

Table 3: Pairs considered most alike by ART and SNA on McCallum email. All pairs produced by the ART model are accurately quite similar. This is not so for the top SNA pairs. Many users are considered similar by SNA merely because they appear in the corpus mostly sending email only to McCallum. However, this causes people with very different roles to be incorrectly declared similar—such as McCallum's spouse and the JMLR editor.

on the other hand, ART more correctly designates a low role similarity for this pair—in fact, ART assigns low similarity between Geaconne and all others in the matrix, which is appropriate because she is the only executive assistant in this small sample of Enron employees.

Another interesting pair of people is Blair (user 4) and Watson (user 14). ART predicts them to be role-similar, while the SNA and AT models do not. ART's prediction seems more appropriate since Blair worked on "gas pipeline logistics" and Watson worked on "pipeline facility planning", two very similar jobs.

McCarty, a Vice-President and CTO in the Division, also highlights differences between the models. The ART model puts him closest to Horton (user 5), who was President of the Division. AT predicts that he is closest to Rapp (user 12), who was merely a lawyer that reviewed business agreements, and also close to Harris (user 15), who was only a mid-level manager.

Using ART in this way emphasizes role similarity, but not group membership. This can be seen by considering Thomas (user 3, an energy futures trader), and his relation to both Rapp (user 12, the lawyer mentioned above), and Lokey (user 16, a regulatory affairs manager). These three people work in related areas, and both ART and AT fittingly indicate a role similarity between them, (ART marginally more so than AT). On the other hand, SNA emphasizes *group memberships* rather than role similarity by placing users 1 through 3 in a rather distinct block structure; they are the only three people in this matrix who were not members of the Enron Transwestern Division group, and these three exchanged more email with each other than with the people of the Transwestern Division. In pending work we are developing a model that integrates both ART and SNA metrics to jointly model both role and group memberships.

Based on the above examples, and other similar examples, we posit that the ART model is more appropriate than the SNA and AT in predicting role similarity.

We thus would claim that the ART model is clearly better than the SNA model in predicting role-equivalence between users, and somewhat better than the AT model in this capacity.

We also carried out this analysis with the personal email for McCallum to further validate the difference between the ART and SNA predictions. There are 825 users in this email corpus. Table 3 shows the closest pairs, as calculated by the ART model and SNA model.

The difference in quality between the ART and SNA halves of the table is striking.

Almost all the pairs predicted by the ART model look reasonable while many of those predicted by SNA are the opposite. For example, ART matches editor and reviews, two email addresses that send messages managing journal reviews. User mike and mikem are actually two different email addresses for the same person. Most other correferent email addresses were pre-collapsed by hand during preprocessing; here ART has pointed out out a mistaken omission, indicating the potential for ART to be used as a helpful component of an automated coreference system. Users aepshtey and smucker were students in a class taught by McCallum. Users coe, laurie and kate are all UMass CS Department administrative assistants; they rarely send email to each other, but they write about similar things. User ang is Andrew Ng from Stanford; joshuago is Joshua Goodman of Microsoft Research; they are both on the organizing committee of a new conference along with McCallum.

On the other hand, the pairs declared most similar by the SNA model are mostly extremely poor. Most of the pairs include donna, and indicate pairs of people who are similar only because in this corpus they appeared mostly sending email only to McCallum, and not others. User donna is McCallum's spouse. Other pairs are more sensible. For example, aepshtey, smucker and rasmith were all students in McCallum's class. User elm is Erik Learned-Miller who is correctly indicated as similar to editor since he is the Production Editor for the Journal of Machine Learning Research.

To highlight the difference between the SNA and ART predictions, we present Table 4, which was obtained by using both ART and SNA to rank the pairs of people by similarity, and then listing the pairs with the highest rank *differences* between the two models. These are pairs that SNA indicated were different, but ART indicated were similar. In every case, there are role similarities between the pairs.

User Pair	*Description*
editor reviews	Both journal editors
jordan mccallum	Both ML researchers
mccallum vanessa	A grad student working in IR
croft mccallum	Both UMass faculty, working in IR
mccallum stromsten	Both ML researchers
koller mccallum	Both ML researchers
dkulp mccallum	Both UMass faculty
blei mccallum	Both ML researchers
mccallum pereira	Both ML researchers
davitz mccallum	Both working on an SRI project

Table 4: Pairs with the highest rank difference between ART and SNA on McCallum email. The traditional SNA metric indicates that these pairs of people are different, while ART indicates that they are similar. There are strong relations between all pairs.

5 Role-Author-Recipient-Topic Models

To better explore the roles of authors, an additional level of latent variable can be introduced to explicitly model roles. Of particular interest is capturing the notion that a person can have multiple roles simultaneously —- a person can be both a professor and a hiker. Each role is associated with a set of topics, and these topics may overlap. For example, professors' topics may prominently feature research, meeting times, grant proposals, and friendly relations; hikers topics may prominently feature mountains, climbing equipment, and also meeting times and friendly relations.

We incorporate into the model a new set of variables that take on values indicating role, and term this augmented model the Role-Author-Recipient-Topic (RART) model. In RART, authors, roles, and message contents are modeled simultaneously. Each author has a multinomial distribution over roles. Authors and recipients are mapped to a role assignments, and then a topic is selected based on these roles. Thus we have a clustering model, in which appearances of topics are the underlying data, and sets of correlated topics gather together clusters that indicate role. Each sender-role and recipient-role pair has a multinomial distribution over topics, and each topic has a multinomial distribution over words.

As shown in Figure 4, different strategies can be employed to incorporate the "role" latent variables. First in RART1, role assignments can be made separately for each word in a document. This model represents that a person can change role during the course of the email message. In RART2, on the other hand, a person chooses one role for the duration of the message. Here each recipient of the message selects a role assignment, and then for each word, a recipient (recipient-role) is selected on which to condition the selection of topic. In RART3, the recipients together result in the selection of a common, shared role, which is used to condition the selection of every word in the message. This last model may help capture the fact that a person's role may depend on the other recipients of the message, but also restricts all recipients to a single role.

We describe the generative process of RART1 in this paper in detail, and leave the other two for subsequent work. In its generative process for each message, an author, a_d, and a set of recipients, $\mathbf{r}_d$, are observed. To generate each word, a recipient, x, is chosen at uniform from $\mathbf{r}_d$, and then a role g for the author, and a role h for the recipient x are chosen from two multinomial role distributions ψ_{a_d} and ψ_x, respectively. Next, a

topic z is chosen from a multinomial topic distribution $\theta_{g,h}$, where the distribution is specific to the author-role recipient-role pair (g, h). Finally, the word w is generated by sampling from a topic-specific multinomial distribution ϕ_z.

In the RART1 model, for a particular message d, given the hyperparameters α, β and γ, the author a_d, and the set of recipients $\mathbf{r}_d$, the joint distribution of an author mixture θ, a role mixture ψ, a topic mixture ϕ, a set of N_d recipients $\mathbf{x}_d$,a set of N_d sender roles $\mathbf{g}_d$, a set of N_d recipient roles $\mathbf{h}_d$, a set of N_d topics $\mathbf{z}_d$ and a set of N_d words $\mathbf{w}_d$ is given by:

$$p(\theta, \phi, \psi, \mathbf{r}_d, \mathbf{g}_d, \mathbf{h}_d, \mathbf{z}_d, \mathbf{w}_d | \alpha, \beta, \gamma, a_d, \mathbf{r}_d) = p(\psi|\gamma)p(\theta|\alpha)p(\phi|\beta) \prod_{n=1}^{N_d} \Big(p(x_{dn}|\mathbf{r}_d)p(g_{dn}|a_d)p(h_{dn}|x_{dn}) p(z_{dn}|\theta_{g_{dn},h_{dn}})p(w_{dn}|\phi_{z_{dn}}) \Big)$$

Integrating over ψ, θ and ϕ, and summing over $\mathbf{x}_d$, g_d, h_d and $\mathbf{z}_d$, we get the marginal distribution of a document:

$$p(\mathbf{w}_d|\alpha, \beta, \gamma, a_d, \mathbf{r}_d) = \iiint p(\psi|\gamma)p(\theta|\alpha)p(\phi|\beta) \prod_{n=1}^{N_d} \sum_{x_{dn}} \sum_{g_{dn}} \sum_{h_{dn}} \sum_{z_{dn}} \Big(p(x_{dn}|\mathbf{r}_d)p(g_{dn}|a_d)p(h_{dn}|x_{dn}) p(z_{dn}|\theta_{g_{dn},h_{dn}})p(w_{dn}|\phi_{z_{dn}}) \mathrm{d}\psi \mathrm{d}\phi \mathrm{d}\theta \Big)$$

Finally, we take the product of the marginal probabilities of single documents, and the probability of a corpus is:

$$p(\mathbf{D}|\alpha, \beta, \gamma, \mathbf{a}, \mathbf{r}) = \prod_{d=1}^{D} p(\mathbf{w}_d|\alpha, \beta, \gamma, a_d, \mathbf{r}_d)$$

To perform inference on RART models, the Gibbs sampling formulae can be derived in a similar way as in Section 2.1, but in a more complex form.

6 Experimental Results with RART

No significant experiments have been conducted on RART models. Based upon our preliminary experimental results with the RART model, properly setting the smoothing parameters is crucial. To make inference more efficiently, we can do inference in distinct parts. For example, because we introduce two additional latent variables (author role and recipient role), the sampling procedure at each iteration is significantly more complicated. One strategy we have found useful is that we can train an ART model first, and use this to fix the topic assignments for each word token. At the next stage, we treat topic as observed, and in this way the RART model can be trained more simply. Although such a strategy may not be recommended for arbitrary graphical models, we feel this is reasonable because we find that a single sample from Gibbs sampling on the ART model yields useful results.

7 Conclusions

We have presented the Author-Recipient-Topic model, a Bayesian network for social network analysis that discovers discussion topics conditioned on the sender-recipient relationships in a corpus of messages. To the best of our knowledge, this model combines for the first time the directionalized connectivity graph from social network analysis with the clustering of words to form topics from probabilistic language modeling.

The model can be applied to discovering topics conditioned on message sending relationships, clustering to find social roles, and summarizing and analyzing large bodies of message data. The model would form a useful component in systems for routing requests, expert-finding, message recommendation and prioritization, and understanding the interactions in an organization in order to make recommendations about improving organizational efficiency.

Additional work on the Role-Author-Recipient-Topic (RART) and other models that explicitly capture roles and groups is ongoing.

Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval, the National Science Foundation under NSF grant #IIS-0326249, and by the Defense Advanced Research Projects Agency, through the Department of the Interior, NBC, Acquisition Services Division, under contract #NBCHD030010.

References

Adamic, L., & Adar, E. (2004). How to search a social network. http://arXiv.org/abs/cond-mat/0310120.

Albert, R., & Barabási, A.-L. (2002). Statistical mechanics of complex networks. *Reviews of Modern Physics*, *74*, 47–97.

Blei, D. M., Ng, A. Y., & Jordan, M. J. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Carley, K. (1991). A theory of group stability. *American Sociological Review*, *56*, 331–354.

Carley, K. (1997). Extracting team mental models through textual analysis. *Journal of Organizational Behavior*, *18*, 533–538.

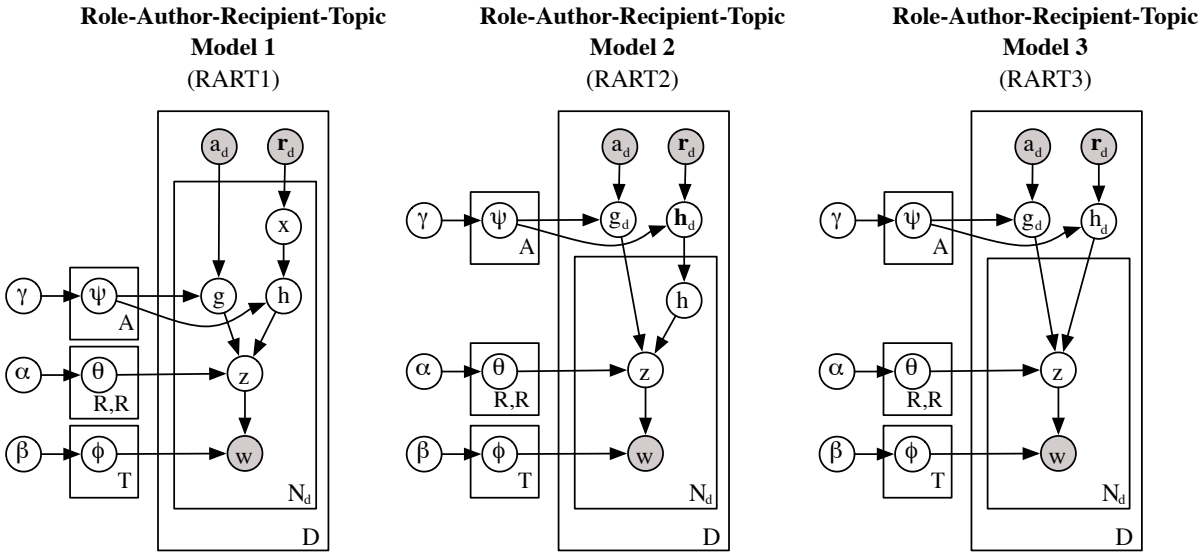


Figure 4: Three possible variants for the Role-Author-Recipient-Topic (RART) model.

Carley, K. (2003). Dynamic network analysis. *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers* (pp. 133–145). National Research Council.

Diesner, J., & Carley, K. (2004a). Automap1.2 – extract, analyze, represent, and compare mental models from texts. CASOS Technical Report, CMU-ISRI-04-100.

Diesner, J., & Carley, K. M. (2004b). Using network text analysis to detect the organizational structure of covert networks. *In Proceedings of the North American Association for Computational Social and Organizational Science (NAACSOS) Conference.*

Diesner, J., Lewis, E., & Carley, K. (2003). How you code matters: How coding techniques and choices influence results from automated text analysis. CASOS working paper.

Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topis. *Proceedings of the National Academy of Sciences* (pp. 5228–5235).

Holland, P., Laskey, K. B., & Leinhardt, S. (1983). Stochastic blockmodels: Some first steps. *Social Networks*, *5*, 109–137.

Kleinberg, J. (2000). Navigation in a small world. *Nature*, *406*, 845.

Krackhardt, D., & Carley, K. M. (1998). A pcans model of structure in organization. *In Proceedings of the 1998 International Symposium on Command and Control Research and Technology.* Monterray, CA.

Lorrain, F., & White, H. C. (1971). The structural equivalence of individuals in social networks. *Journal of Mathematical Sociology*, *1*, 49–80.

McCallum, A. (1999). Multi-label text classification with a mixture model trained by EM. *AAAI Workshop on Text Learning.*

Minka, T., & Lafferty, J. (2002). Expectation-propagation for the generative aspect model. *In Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence.* New York: Elsevier.

Rosen-Zvi, M., Griffiths, T., Steyvers, M., & Smyth, P. (2004). The author-topic model for authors and documents. *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence.* Banff, Alberta, Canada.

Shetty, J., & Adibi, J. (2004). *The Enron email dataset database schema and brief statistical report* (Technical Report). Information Sciences Institute.

Steyvers, M., Smyth, P., Rosen-Zvi, M., & Griffiths, T. (2004). Probabilistic author-topic models for information discovery. *The Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

Teh, Y. W., Jordan, M. I., Beal, M. J., & Blei, D. M. (2004). *Hierarchical dirichlet processes* (Technical Report). UC Berkeley Statistics.

Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications.* Cambridge University Press.

Watts, D. J. (2003). *Six degrees: The science of a connected age.* Norton.

Watts, D. J., Dodds, P. S., & Newman, M. E. J. (2002). Identify and search in social networks. *Science*, *296*, 1302–1305.

Wolfe, A. P., & Jensen, D. (2003). Playing multiple roles: Discovering overlapping roles in social networks. *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.*

Wu, F., Huberman, B. A., Adamic, L. A., & Tyler, J. R. (2003). Information flow in social groups. http://arXiv.org/abs/cond-mat/0305305.

Email Surveillance Using Nonnegative Matrix Factorization

Michael W. Berry* Murray Browne†

February 25, 2005

Abstract

In this study, we apply a non-negative matrix factorization approach for the extraction and detection of concepts or topics from electronic mail messages. For the publicly released Enron electronic mail collection, we encode sparse term-by-message matrices and use a low rank non-negative matrix factorization algorithm to preserve natural data non-negativity and avoid subtractive basis vector and encoding interactions present in techniques such as principal component analysis. Results in topic detection and message clustering are discussed in the context of published Enron business practices and activities, and benchmarks addressing the computational complexity of our approach are provided. The resulting basis vectors and matrix projections of this approach can be used to identify and monitor underlying semantic features (topics) and message clusters in a general or high-level way without the need to read individual electronic mail messages.

Keywords: electronic mail, Enron collection, nonnegative matrix factorization, surveillance, topic detection, constrained least squares.

1 Background

One of the by-products of the Federal Energy Regulatory Commission's (FERC) investigation of Enron was the vast amount of information (electronic mail messages, phone tapes, internal documents) collected towards building a legal case against the global energy corporation. As a matter of public record, this information which initially contained over 1.5 million electronic mail (email) messages was originally posted on FERC's web site [9]. However the original set suffered from document integrity problems and attempts were made to improve the quality of the data and remove some of the sensitive and irrelevant private information. Dr. William Cohen of Carnegie Mellon University took the lead in distributing this improved corpus – known as the Enron Email Sets. The latest version of the Enron Email Sets[1] (dated – March 2, 2004) contains 517,431 email messages of 150 Enron employees covering a period from December 1979 through February 2004 with the majority of messages spanning the three years: 1999, 2000, and 2001. It includes messages of some of the top executives of Enron management personnel including founder and Chief Executive Officer (CEO) Ken Lay, president and Chief Operating Officer (COO) Jeff Skilling, and head of trading and later COO, Greg Whalley. Other top executives who played major roles in the day-to-day operations of the corporation are represented as well. They include: Louise Kitchen who developed the Enronline, the corporations in-house trading system, Vince Kaminski head of research, Richard Sanders leader of Enron North America's litigation department and Steve Kean Executive Vice President and Chief of Staff.

In addition to operational logistics of being America's seventh largest company, Enron was faced with many ongoing crises. One involved Enron's development of the Dabhol Power Company (DPC) in the Indian state of Maharashtra, an endeavor awash in years of logistical and political problems. Then there was the deregulation of the California energy market, which led to rolling blackouts during the summer of 2000 - a situation that Enron (and other energy companies) took advantage of financially. By the fall of 2001, Enron's combination of greed, overspeculation, and deceptive accounting practices snowballed into an abrupt collapse. A last minute merger with the Dynegy energy company fell through and Enron filed for Chapter 11 bankruptcy on December 2, 2001 [18]. As expected, The Enron Email Sets reflect this business world ranging from corporate memos to fantasy football picks. The challenge was how to classify this information in a meaningful way.

In Section 2 we discuss one mathematical approach to the extraction of *features* from subcollections of Enron electronic messages – non-negative matrix factoriza-

[1] http://www-2.cs.cmu.edu/∼enron

tion. Building upon previous work in topic detection on benchmark collections (with human curated classifications) [21], we apply this *parts*-based factorization approach to topic detection and monitoring of electronic mail messages. Such an application could facilitate the design of future *surveillance* systems in which topics of electronic mail discussions are identified (without literally reading messages) and tracked over time. In Section 3, we describe two particular subsets of the Enron collection that were parsed and analyzed for topic tracking. Section 4 provides illustrations of successful topic detection along with caveats to the use of non-negative factorization in this context. Tracking one particular year (2001) of an Enron subcollection shows which corporate deals and activities dominated the corporation prior to and during its collapse. There is also a discussion of how to distinguish different types of electronic messages. Finally, we conclude our findings and suggest future modeling of the Enron collection in Section 5.

2 Non-Negative Matrix Factorization

Non-negative matrix factorization (NMF) has recently been shown to be a very useful technique in approximating high dimensional data where the data are comprised of non-negative components. In a seminal paper published in *Nature* [15], Lee and Seung proposed the idea of using NMF techniques to find a set of basis functions to represent image data where the basis functions enable the identification and classification of intrinsic "parts" that make up the object being imaged by multiple observations. They showed that NMF facilitates the analysis and classification of data from image or sensor articulation databases made up of images showing a composite object in many articulations, poses, or observation views. They also found NMF to be a useful tool in text data mining. In the past few years, several papers have discussed NMF techniques and successful applications to various databases where the data values are non-negative, e.g., [7, 11, 12, 13, 16, 17, 22].

More generally, matrix factorization techniques in data mining fall under the category of vector space methods. Very often databases of interest lead to a very high dimensional matrix representation. Low-rank factorizations not only enable the user to work with reduced dimensional models, they also often facilitate more efficient statistical classification, clustering and organization of data, and lead to faster searches and queries for patterns or trends, e.g., Berry, Drmač, and Jessup [4]. Recently, Xu et al [23] demonstrated that NMF-based indexing outperforms traditional vector space approaches to information retrieval (such as latent semantic indexing) for document clustering on a few benchmark test collections.

NMF is a vector space method used to obtain a representation of data using non-negativity constraints. These constraints can lead to a parts-based representation because they allow only additive, not subtractive, combinations of the original data. This is in contrast to techniques for finding a reduced dimensional representation based on singular value decomposition-type methods such as principal component analysis (PCA) [14]. One major problem with PCA is that the basis vectors have both positive and negative components, and the data are represented as linear combinations of these vectors with positive and negative coefficients. In many applications, however, the negative components contradict physical realities. In particular, term frequencies in text mining are non-negative. In this paper, we survey some popular computational approaches (and their complexities) for NMF in the context of document clustering applications, and demonstrate the use of a *new* hybrid NMF method that can enforce smoothness (or sparsity) constraints on the resulting factor matrices.

2.1 Optimization Problem Given a collection of electronic mail messages expressed as an $m \times n$ term-by-message matrix X, where each column is an m-dimensional non-negative vector of the original collection (n vectors), the standard NMF problem is to find two new reduced-dimensional matrices W and H, in order to approximate the original matrix X by the product WH in terms of some metric. Each column of W contains a *basis vector* while each column of H contains the *weights* needed to approximate the corresponding column in X using the basis from W. The dimensions of matrices W and H are $m \times r$ and $r \times n$, respectively. Usually, the number of columns in the new (basis) matrix W is chosen so that $r \ll n$. Here, the choice of r is generally application dependent, and may also depend upon the characteristics of the particular corpus or database [11].

The usual approach to the NMF problem is to approximate X by computing a pair W and H to minimize the Frobenius norm of the difference $X - WH$. Mathematically, the problem can be stated as follows: Let $X \in R^{m \times n}$ be a data matrix of non-negative entries. Let $W \in R^{m \times r}$ and $H \in R^{r \times n}$ for some positive integer $r < n$. The objective is then to solve the optimization problem

$$\min_{W,H} \|X - WH\|_F^2, \tag{2.1}$$

subject to $W_{ij} \geq 0$ and $H_{ij} \geq 0$ for each i and j.

Of course the matrices W and H are generally not unique. Conditions resulting in uniqueness in the special case of equality, $X = WH$, have been recently studied by Donoho and Stodden [7], using cone theoretic

techniques (See also Chapter 1 in Berman and Plemmons [1]). Algorithms designed to approximate X by solving the minimization problem (2.1) generally begin by initial estimates of the matrices W and H, followed by alternating iterations to improve these estimates.

To explain the non-negative matrix factorization approach used in this study, we briefly review previous methods discussed in the literature.

2.2 Multiplicative Method. A non-negative matrix factorization algorithm of Lee and Seung [15] is based on multiplicative update rules of W and H. We call this scheme the *multiplicative method*, and denote it by **MM**. A formal statement of the method is given below:

MM Algorithm (Lee and Seung)

1. Initialize W and H with non-negative values, and scale the columns of W to unit norm.
2. Iterate for each c, j, and i until convergence or after k iterations:
   (a) $H_{cj} \leftarrow H_{cj} \frac{(W^T X)_{cj}}{(W^T W H)_{cj} + \epsilon}$
   (b) $W_{ic} \leftarrow W_{ic} \frac{(X H^T)_{ic}}{(W H H^T)_{ic} + \epsilon}$
   (c) Scale the columns of W to unit norm.

Clearly the approximations W and H remain non-negative during the updates. It is generally best to update W and H *simultaneously*, instead of updating each matrix fully before the other. In this case, after updating a row of H, we update the corresponding column of W. In the implementation described in [21], a small positive quantity, say the square root of the machine precision, should be added to the denominators in the approximations of W and H at each iteration step. Setting $\epsilon = 10^{-9}$ will typically suffice.

It is often important to normalize the columns of X in a pre-processing step, and in the algorithm to normalize the columns of the basis matrix W at each iteration. In this case we are optimizing on a unit hypersphere, as the column vectors of W are effectively mapped to the surface of a hypersphere by the repeated normalization.

The computational complexity of Algorithm MM can be shown to be $O(rmn)$ operations per iteration. Additional data (e.g., new electronic mail messages) can either be added directly to the basis matrix W along with a minor modification of H, or else if r is fixed, then further iterations can be applied starting with the current W and H as initial approximations.

Lee and Seung [16] proved that under the MM update rules the distance $\|X - WH\|_F^2$ is monotonically non-increasing. In addition it is invariant if and only if W and H are at a stationary point of the objective function in Eq. (2.1). From the viewpoint of nonlinear optimization, the algorithm can be classified as a diagonally-scaled gradient descent method [11]. Lee and Seung [15] have also provided an additive algorithm. Both the multiplicative and additive algorithms are related to expectation-maximization approaches used in image processing computations such as image restoration, e.g., [20].

2.3 Enforcing Statistical Sparsity. Hoyer [12] has suggested a novel non-negative sparse coding scheme based on ideas from the study of neural networks, and the scheme has been applied to the decomposition of databases into independent feature subspaces by Hyvärinen and Hoyer [13]. Hoyer's method [12] has the important feature of enforcing a statistical sparsity for the weight matrix H, thus enhancing the parts-based representation of the data in W.

Mu, Plemmons and Santago [19] propose a regularization approach that, like Hoyer's method, can be used to enforce statistical sparsity of the weight matrix H. This approach uses a so-called point count regularization scheme in the computations that penalizes the *number* of nonzero entries in H, rather than $\sum_{ij} H_{ij}$, as proposed by Hoyer. Sparsity often leads to a basis representation in W that better represents parts or features of the corpus defined by X [21].

2.4 A Hybrid NMF Approach. We use a hybrid algorithm for NMF that combines some of the better features of available methods. As discussed in [21], the multiplicative algorithm approach can be used to compute an approximation to the basis matrix W at each iterative step. This computation is essentially a matrix version of the gradient descent optimization scheme mentioned earlier. Secondly, we compute the weight matrix H using a constrained least squares (CLS) model as the metric. The purpose is to penalize non-smoothness and non-sparsity in H. This approach bears similarity to those of Hoyer and Mu, Plemmons and Santago. The CLS model is related to the least squares Tikhonov regularization technique commonly used in image restoration [20]. As presented in [21], the algorithm, referred to as **GD-CLS** for *gradient descent with constrained least squares*, is given below:

Algorithm for GD-CLS [21]

1. Initialize W and H with non-negative values, and

scale the columns of W to unit norm.

2. Iterate until convergence or after k iterations:

   (a) $W_{ic} \leftarrow W_{ic} \frac{(XH^T)_{ic}}{(WHH^T)_{ic} + eps}$, for c and i

   (b) Rescale the columns of W to unit norm.

   (c) Solve the constrained least squares problem:

$$\min_{H_j} \{\|X_j - WH_j\|_2^2 + \lambda \|H_j\|_2^2\},$$

   where the subscript j denotes the j^{th} column, for $j = 1, \ldots, m$. Any negative values in H_j are set to zero. The parameter λ is a regularization value that is used to balance the reduction of the metric

$$\|X_j - WH_j\|_2^2$$

   with enforcement of smoothness and sparsity in H.

As done in Algorithm MM, we use a small positive parameter ϵ to avoid dividing by zero or very small numbers and enhance stability in the computations for W in Step 2(a). The numerical approach for solving the constrained least squares problem in Step 2(c) for the columns H_j of H makes use of an algorithm similar to one described in [20] for regularized least squares image restoration.

3 Electronic Mail Subcollections

We have recently tested the effectiveness of the **GD-CLS** algorithm for computing the non-negative matrix factorization of term-by-message matrices derived from the Enron corpus. These matrices were derived from the creation and parsing of two subcollections: INBOX and PRIVATE . Our rationale for creating these two particular subcollections of the raw Enron collection is that INBOX would reflect a standard (perhaps untarnished) repository of all incoming messages to an Enron employee, and PRIVATE would represent personal classifications of messages originally posted to a user's *inbox* folder and then copied or moved to discriminated folders. Although no attempt is made with the **GD-CLS** algorithm to guarantee that messages of the same folder (user-assigned topic) are spanned by similar feature vectors, the semantic interpretation of feature vectors (using the components of the W and H factors) is greatly improved when taking into account the user's original clustering of messages, i.e., message directory path.

3.1 Message Parsing. The INBOX subcollection is comprised all emails contained in the *inbox* folder of all 150 users (or subdirectories) in the raw dataset. Using a 495-term *stoplist* of unimportant terms (or words), the GTP software environment [10] extracted $80,683$ terms from $44,872$ electronic messages. This subcollection reflects 8.7% of the $517,431$ messages in the *raw* Enron collection. With the same stoplist and parsing all users' mail directories with the exception of *all_documents*, *calendar*, *contacts*, *deleted_items discussion_threads*, *inbox*, *notes_inbox*, *sent*, *sent_items*, and *_sent_mail*, GTP extracted $92,133$ terms from $65,033$ messages (29.1% of the raw collection) to define the PRIVATE subcollection. In order to simulate the tracking of topics through an eventful year, say 2001, in the corporate life of Enron, we also created twelve smaller subsets of the PRIVATE subcollection. As depicted in Table 1, all messages sent in a particular month of 2001 were parsed to create twelve separate dictionaries (or sets of terms). As will be discussed in Section 4.3, we use these smaller collections to track topics throughout the year with no accumulation of dictionaries, that is, we apply **GD-CLS** to each corresponding term-by-message matrix separately and extract message clusters (topics) independently. An alternative approach for topic tracking through time would be to update the non-negative matrix factorization with each new month's set of messages. Methods for the efficient updating of Eq. (4.3) are now under consideration (see [21]) and are not in the scope of this work.

In creating the subcollections, all permissible folders are elgible for parsing (no threshold on the number of messages applied) and all message headers are left intact for GTP to process. All terms (or keywords) comprising the resulting dictionary are required to occur at least twice (globally) across the particular subcollection and in two or more messages. In order to define meaningful term-to-message associations for concept discrimination, term weighting is used in the generation of all term-by-message matrices.

3.2 Term Weighting. As explained in [2], a collection of n messages indexed by m terms (or keywords) can be represented as a $m \times n$ term-by-message matrix $X = [x_{ij}]$. Each element or component x_{ij} of the matrix X defines a *weighted* frequency at which term i occurs in message j [3]. We can define

$$x_{ij} = l_{ij} g_i d_j, \tag{3.2}$$

where l_{ij} is the local weight for term i occurring in message j, g_i is the global weight for term i in the subcollection, and d_j is a document normalization factor which specifies whether or not the columns of X (i.e., the documents) are normalized (i.e., have unit length).

Table 1: Counts of messages from the PRIVATE subcollection that were sent on each month of 2001. The corresponding number of terms parsed for each monthly subset is denoted as well..

Month	Messages	Terms
Jan	3,621	17,888
Feb	2,804	16,958
Mar	3,525	20,305
Apr	4,273	24,010
May	4,261	24,335
Jun	4,324	18,599
Jul	3,077	17,617
Aug	2,828	16,417
Sep	2,330	15,405
Oct	2,821	20,995
Nov	2,204	18,693
Dec	1,489	8,097

Let f_{ij} be the number of times (frequency) that term i appears in message j, and define $p_{ij} = f_{ij}/\sum_j f_{ij}$. Two possible definitions for x_{ij} in Eq. (3.2) are given by Table 2. We use **txx** and **lex** to refer to simple (term) frequency and log-entropy term weighting, respectively.

Table 2: Term weighting schemes used in the parsing of the INBOX and PRIVATE subcollections. No message normalization is applied so that $d_j = 1$ in Eq. (3.2) and base 2 logarithms should be assumed.

	Weighting Component	
Name	Local	Global
txx	Term Frequency $l_{ij} = f_{ij}$	None $g_i = 1$
lex	Logarithmic $l_{ij} = \log(1 + f_{ij})$	Entropy [8] $g_i = 1+$ $(\sum_j p_{ij} log(p_{ij}))/\log n)$

4 Observations and Results

Figures 1 and 2 illustrate the different cluster sizes obtained from the non-negative matrix factorization of the term-by-message matrix X associated with the PRIVATE collection with log-entropy and simple term frequency weighting, respectively. Here, we approximate

Table 3: **GD-CLS** benchmarks for computing the non-negative factorization in Eq. (4.3), where X is generated from either the INBOX and PRIVATE electronic mail collections. Exactly 50 clusters (topics), which is also the column dimension of the W matrix and row dimension of the H matrix, are generated, and λ is the regularization parameter controlling the sparsity of the matrix H. Time is elapsed CPU time in seconds on a 450MHz (Dual) UltrasSPARC-II processor for 100 iterations of **GD-CLS**.

Collection	Mail Messages	Dictionary Terms	λ	Time (sec.)
INBOX	44,872	80,683	0.1	1,471
			0.01	1,451
			0.001	1,521
PRIVATE	65,031	92,133	0.1	51,489
			0.01	51,393
			0.001	51,562

the $92,133 \times 65,031$ (sparse) matrix X via

$$X \simeq WH = \sum_{k=1}^{50} W_k H^k \ , \tag{4.3}$$

where W and H are $92,133 \times 50$ and $50 \times 65,031$, respectively, non-negative matrices. W_k denotes the kth column of W, H^k denotes the kth row of the matrix H, and $r = 50$ factors or parts are produced. Clearly, the non-negativity of W and H facilitate a parts-based representation of the matrix X whereby the basis (column) vectors of W or W_k combine to approximate the original columns (messages) of the sparse matrix X. The outer product representation of WH in Eq. (4.3) demonstrates how the rows of H or H^k essentially specify the weights (scalar multiples) of each of the basis vectors needed for each of the 50 parts of the representation. As described in [15], we can interpret the semantic feature represented by a given basis vector W_k by simply sorting (in descending order) its 92,133 elements and generating a list of the corresponding dominant terms (or keywords) for that feature. In turn, a given row of H having n elements (i.e., H^k) can be used to reveal messages sharing common basis vectors W_k, i.e., similar semantic features or meaning. The columns of H, of course, are the projections of the columns (messages) of X onto the basis spanned by the columns of W. The best choice for the number of parts r (or column rank of W) is certainly problem-dependent or corpus-dependent in this context. However, as discussed in [21] for

standard topic detection benchmark collections (with human-curated document clusters) the accuracy of **GD-CLS** for document clustering degrades as the rank r increases or if the sizes of the clusters become greatly imbalanced. Further investigations into the effects of message clustering with larger ranks (beyond 50) are planned.

The association of features (i.e., feature vectors) to the electronic mail messages is accomplished by the nonzeros of each H^k which would be present in the kth *part* of the approximation to X in Eq. (4.3). Each part (or span of W_k) can be used to classify the messages so the sparsity of H greatly affects the diversity of topics with which any particular semantic feature can be associated. In Figures 1 and 2, we show the number of nonzero elements in each H^k of magnitude greater than $row_{max}/10$ for three different choices of λ (namely 0.001, 0.01, and 0.1) in the **GD-CLS** algorithm. Using the rows of the H matrix and a threshold on the nonzero elements to cluster messages, we obtain quite a wide range of cluster sizes. As λ increases, we do not uniformly see a decrease in the cluster sizes as might be expected (due to an expected increase in the sparsity of H). However for most clusters (or rows of H) there is some reduction in the number of elements exceeding the $row_{max}/10$ threshold. The increased sparsity in H for larger values of λ is also reflected in the elapsed CPU times shown in Table 3. For a more thorough assessment of the reduction in statistical sparsity of the matrix H generated by the **GD-CLS** algorithm see [21].

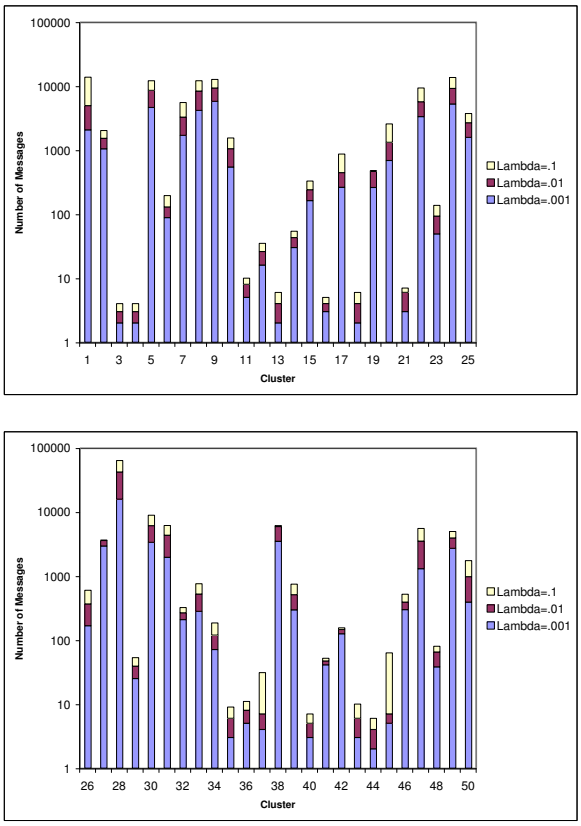


Figure 1: Size of clusters (number of electronic mail messages) produced by the **GD-CLS** algorithm for the PRIVATE collection. Log-entropy term-weighting is used. Three instances of the regularization parameter ($\lambda = 0.001, 0.01, 0.1$) for controlling the sparsity of the H matrix factor are shown in each graph.

4.1 Topic Extraction. Tables 4 and 5 illustrate some of the extracted topics (i.e., message clusters) as evidenced by large components in the same row of the matrix H (or H^k) generated by **GD-CLS** for the sparse term-by-message matrix associated with the PRIVATE subcollection. The terms corresponding to the 10-largest elements of the particular feature (or part) k are also listed to explain and derive the context of the topic. By feature, we are referring to the k-th column of the matrix factor W or W_k in Eq. (4.3), of course. The seven topics reflected in Table 4 do occur in the parts-based factorization of the matrix X regardless of whether **lex** or **txx** weighting (see Section 3.2) are used by the GTP software environment. The three topics shown in Table 5, however, were extracted only from the use of **txx** weighting. It is interesting to note that the use of a single term-weighting scheme might have a limiting effect on the ability to discern/interpret context of the features produced by a non-negative matrix factorization. Further studies into such effects are needed.

In a perfect email surveillance world, each cluster of terms would point to the documents by a specific topic. Although our experiments did not produce such results for every cluster, they did give some indication of what the particular message collection *was about*. With 50 clusters or features produced by **GD-CLS** and deploying both **lex** and **txx** for different instances of a term-by-message matrix X, we analyzed the ten dominant (in magnitude) terms per feature for clues about the content of the collection. The majority of the cluster terms were too vague or too broad to be meaningful, but each variation did reveal clusters that merited further investigation. These clusters had a tendency to have a few proper nouns – words such as *kitchen* (for Louise Kitchen) or company names such as *dynegy* coupled with other more general terms such as *merger* which in the case of *dynegy* and *merger* would point to documents that were referring to the

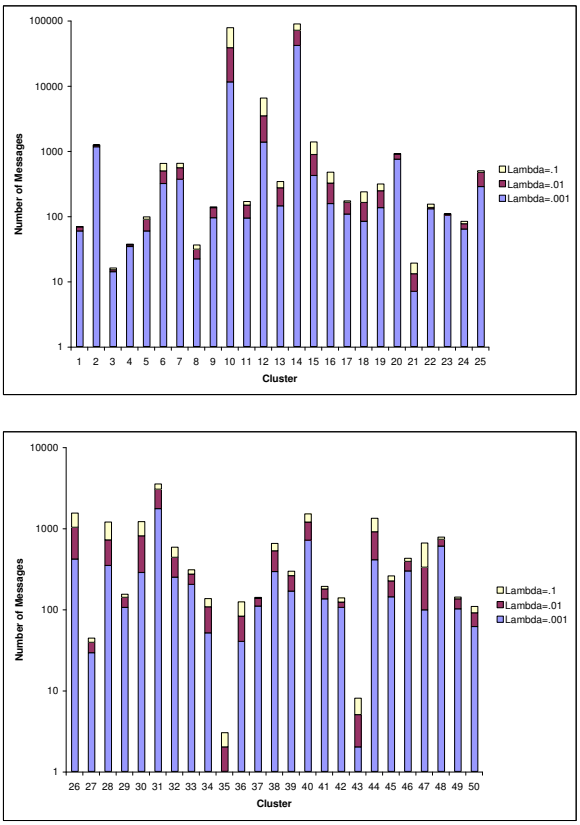


Figure 2: Size of clusters (number of electronic mail messages) produced by the **GD-CLS** algorithm for the PRIVATE collection. Simple term frequency weighting is used. Three instances of the regularization parameter ($\lambda = 0.001, 0.01, 0.1$) for controlling the sparsity of the H matrix factor are shown in each graph.

last minute efforts of Enron to avoid total collapse by merging with the Dynegy corporation. For these type of "meaningful" clusters, we checked to verify that the documents were semantically linked to the terms of the clusters.

The more promising clusters (those that were specific enough to indicate what might be found if one looked at the corresponding documents) were clusters that referred to a median range (say in the hundreds and not thousands) of messages (see Figures 1 and 2). Clusters with only one or several messages were found to be inconclusive. Keep in mind that we are measuring cluster or feature size by the number of row elements in the matrix H with magnitude greater than a specified tolerance (which is $row_{max}/10$ for this study). Conversely, clusters representing thousands of mail messages were unmanageable.

Table 4: Sample clusters (topics) identified by the rows of H or H^k produced by the non-negative matrix factorization (with $\lambda = 0.1$) of the term-by-message matrix X associated with the PRIVATE subcollection and **lex** term-weighting. Exactly $r = 50$ feature vectors (W_k) were generated by the **GD-CLS** algorithm. The ten dominant (having values of largest magnitude) terms for each feature vector are listed for each selected feature (k), and those in **boldface** were judged to be the most descriptive. Cluster size reflects the number of row elements in H^k of magnitude greater than $row_{max}/10$.

Feature Index (k)	Cluster Size	Topic Description	Dominant Terms
10	497	California	ca, **cpuc,** gov, **socalgas,** sempra, org, sce, gmssr, aelaw, ci
23	43	Louise Kitchen named top woman by Fortune	evp, **fortune,** britain, woman, **ceo,** avon, fiorinai, cfo, hewlett, packard
26	231	Fantasy football	game, wr, qb, play, rb, season, injury, updated, fantasy, image
33	233	Texas longhorn football newsletter	UT, orange, longhorn[s], texas, true, truorange, recruiting, oklahoma defensive
34	65	Enron collapse	**partnership[s], fastow**, shares, **sec**, stock, shareholder, investors, equity, **lay**
39	235	Emails about India	**dahhol, dpc, india, mseb, maharashtra,** indian, lenders, delhi, foreign, minister
46	127	Enron collapse	dow, debt, reserved, wall, copyright jones, cents, analysts, reuters, spokesman

For example, a cluster with the terms *power, california, electricity, demand* represented 2,253 documents (a similar California cluster had 8,500 messages) which is so general that it is of limited use. With this is mind, we made another pass looking at each of the clusters that represented anywhere from 10 to 500 messages even if their terms were initially seemed vague.

One example of a meaningful cluster that seemed too vague at first was the cluster associated with feature index $k = 23$ (see Table 4). This feature spanned such terms as: *evp, fortune, britain, women, ceo, avon, fiorina, cfo, hewlett*, and *packard*. But the cluster defined by the dominant components of H^{23} was composed of 43 messages and thus merited further investigation. A look at those documents revealed a set of electronic mail messages that referred to Louise Kitchen's selection in *Fortune*'s 2001 List of the Fifty Most Powerful Women in Business. The messages even included congratulatory notes from her Enron colleagues.

Perhaps one of the most revealing clusters of this series of experiments, were the football-related clusters. Not only did the clusters reveal which messages (and their participants) were linked to football, but it was able to differentiate between fantasy football leagues, which are typically associated with professional teams, and the University of Texas Longhorn football team.

In the three topics that were unique to the **txx** weighting, the cluster of messages associated with feature $k = 16$ in Table 5 is merely a list of rampant database error messages that were forwarded to a user. The first cluster in that table ($k = 2$) refers to a series of memos about preparing for a possible investigation from a California state senator and the third cluster ($k = 40$) focuses on various gas and oil contracts.

4.2 Message Size. One problem in working with such a volume of emails is that the clusters can be influenced by news wire feeds and other automatically generated content. When examining the messages of each cluster, a message corresponding to the largest component of H^k was usually a news wire feed. For example, with feature $k = 39$ in Table 4 we find that 4.50 is the highest value associated with any message in the cluster. As expected, checking the message reveals a 1,700-line *Wall Street Journal* news wire article on Dabhol. Component values of H^k for a specific cluster k can also help reveal which messages are news feeds (of little surveillance value) and which messages may be smaller emails with more concise content. For example, in feature $k = 39$, one message identified by a component value (in the k-th row of H or H^k) of just 0.9 is a short message from Vince Kaminski to Jeff Skilling but it belongs in the India topic cluster because the message strategizes about India. The ability to distinguish between large messages such as news feeds and smaller more personal messages can be gauged by the type of term-weighing scheme (e.g., **lex** or **txx**) deployed. See [8] and [2] for more details on specific attempts to take document (or message) length into account for term-weighting.

Ironically, in the early stages of our results assessment it was the prevalence (and frustration) of the large news wire stories in the Enron INBOX subcollection that prompted us to concentrate more on the PRIVATE subcollection. Also, as mentioned earlier, the PRIVATE subcollection of emails from the Enron Email Sets represents a larger portion of the collection of over 65,000 messages as compared to only approximately 45,000 messages for the INBOX subcollection. One could also make the argument that in general the messages comprising PRIVATE subcollection were more important to a Enron employee because he or she had to at least evaluate the content of the messages before categorizing them (i.e., moving them to folders).

Table 5: Selected clusters (topics) identified by the rows of H or H^k produced by the non-negative matrix factorization of the term-by-message matrix X associated with the PRIVATE subcollection and **txx** term-weighting. Exactly $r = 50$ feature vectors (W_k) were generated by the **GD-CLS** algorithm (with $\lambda = 0.1$). The ten dominant (having values of largest magnitude) terms for each feature vector are listed for each selected feature (k), and those in **boldface** were judged to be the most descriptive. Cluster size reflects the number of row elements in H^k of magnitude greater than $row_{max}/10$.

Feature Index (k)	Cluster Size	Topic Description	Dominant Terms
2	13	Dunn investigation; document retention policy	documents, committee, subpoena, intended, **brobeck**, senate, records, recipient, email, section
16	156	Database error messages	database, dbcaps97data, **davis**, unknown, alias, pete, date, bill, mark, error
40	311	Gas contracts	gas, natural, oil, pipeline, contract, storage, **el**, prices, **paso**, daily

4.3 Temporal Monitoring. Because the calendar year 2001 comprised the largest volume of electronic mail of any single year of the Enron subcollections considered, we examined the performance of a rank $r = 50$ non-negative factorization (with **lex** term-weighting and $\lambda = 0.1$) on twelve specific subsets of the PRIVATE subcollection. Namely, we isolated those electronic messages sent in each month of the calendar year 2001. We looked at how previously defined topics such as California, India, the bankruptcy after the Dynegy merger fell through, and both football topics (fantasy and college) were represented on a month to month basis. Figure 3 illustrates how clusters/topics identified by the non-negative matrix factorization can be traced through time. The results were consistent with what might expect given the history of the Enron Corporation in 2001. The year began with California Governor Gray Davis calling for an investigation of Enron in light of the 2000 California Energy crisis and it was an ongoing topic throughout the year. To a lesser degree, the discussion and legal battles involving the Dabhol Power Company were also consistently present throughout the year. Perhaps a more poignant example of how the **GD-CLS**-generated clusters reflect timeliness is with the topic of the Dynegy merger and subsequent bankruptcy of Enron. These clusters came to the forefront in the fourth quarter of the year which coincided with Enron's final attempts in November to save itself by merging with Dynegy. The football clusters also demonstrate the ability of the clusters to reflect chronological events. As one would expect, college football dominated in the fall and fantasy (professional) football came on strong in December. The most noteworthy aspect of the temporal monitoring is that the process even identified a cluster of Texas football messages present in May of 2001 (perhaps reflecting the university's spring football practices).

Although, the **GD-CLS**-derived models were unable to generate clusters of very specific topics (something that would be of great value for email surveillance), the resulting parts-based factorizations do give a sense of "aboutness" to the Enron world of international energy management and some general direction on where specific documents on certain topics may be found.

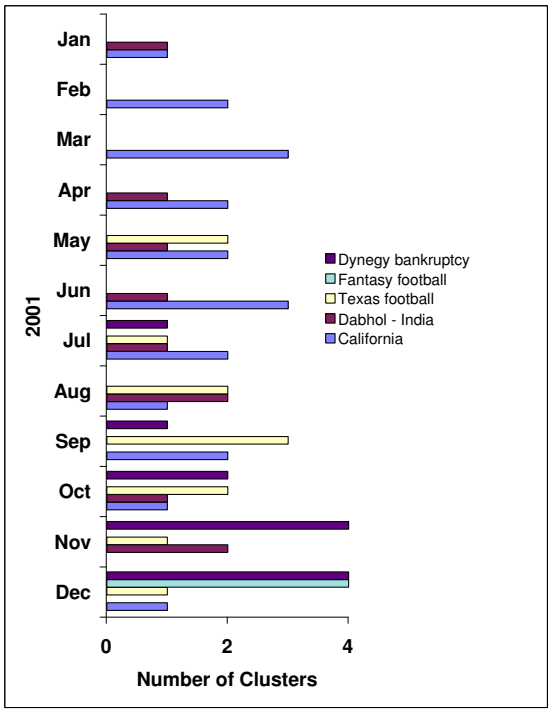


Figure 3: Number of instances of detectable topics for the calendar year 2001 using $r = 50$ features produced by the **GD-CLS** algorithm. Twelve subsets of the PRIVATE collection (one per month) were parsed with each subset comprising all electronic messages sent during the corresponding month of 2001. Log-entropy (or **lex**) term-weighting and the regularization parameter $\lambda = 0.1$ was used for each run of the **GD-CLS** algorithm.

5 Concluding Remarks

We have demonstrated how the **GD-CLS** algorithm for computing the non-negative matrix factorization can be used for extracting and tracking topics of discussion from corporate email. This algorithm effectively computes a parts-based approximation $X \simeq WH$ of a sparse term-by-message matrix X in which the quality of approximation (error reduction) can be enhanced by an enforcement of smoothness and sparsity in the non-negative matrix H. Although little or no information was extracted to potentially expose fraudulent actions or behaviors of Enron employees, we have demonstrated how a parts-based representation of corporate electronic mail (e.g., Enron) can facilitate the *observation* of electronic message discussions without requiring human intervention or the reading of individual messages. Such surveillance enables corporate leaders (say managers or supervisors) to monitor discussions without the need to isolate or perhaps incriminate individual employees. In this way, factors such as company morale, employees' feedback to policy decisions, and extracurricular activities may eventually be tracked.

With respective to the **GD-CLS** algorithm, further work is needed in exploring the effects of different term

weighting schemes (for X) on the quality of the basis vectors W_k. How document (or message) clustering changes with different column ranks in the matrix W should be considered as well.

References

[1] A. Berman and R. Plemmons. *Non-Negative Matrices in the Mathematical Sciences*, SIAM Press Classics Series, Philadelphia, 1994.

[2] M. Berry and M. Browne. *Understanding Search Engines: Mathematical Modeling and Text Retrieval*, SIAM, Philadelphia, 1999.

[3] M. Berry, S. Dumais, and G. O'Brien "Using Linear Algebra for Intelligent Information Retrieval", *SIAM Review*, Vol. 37, No. 4, pp. 573-595, 1995.

[4] M. Berry, Z. Drmač, and E. Jessup. "Matrices, Vector Spaces, and Information Retrieval", *SIAM Review*, Vol. 41, No. 2, pp. 335-362, 1999.

[5] *Concise Columbia Encyclopedia*. Columbia University Press, New York, Second Edition, 1989.

[6] M. Cooper and J. Foote, "Summarizing Video using Non-Negative Similarity Matrix Factorization", *Proc. IEEE Workshop on Multimedia Signal Processing* St. Thomas, US Virgin Islands, 2002.

[7] D. Donoho and V. Stodden. "When does Non-Negative Matrix Factorization Give a Correct Decomposition into Parts?", preprint, Department of Statistics, Stanford University, 2003.

[8] S. Dumais, "Improving the Retrieval of Information from External Sources", *Behavior Research Methods, Instruments, & Computers*, Vol. 23, No. 2, pp. 229-236, 1991.

[9] T. Grieve, "The Decline and Fall of the Enron Empire", *Slate*, October 14, 2003, `http://www.salon.com/news/feature/2003/10/14/enron/index_np.html`.

[10] J.T. Giles, L. Wo, and M.W. Berry. "GTP (General Text Parser) Software for Text Mining", in *Statistical Data Mining and Knowledge Discovery*, H. Bozdogan (Ed.), CRC Press, Boca Raton, (2003), pp. 455-471.

[11] D. Guillamet and J. Vitria. "Determining a Suitable Metric when Using Non-Negative Matrix Factorization", *16th International Conference on Pattern Recognition (ICPR'02)*, Vol. 2, Quebec City, QC, Canada, 2002.

[12] P. Hoyer. "Non-Negative Sparse Coding", *Neural Networks for Signal Processing XII (Proc. IEEE Workshop on Neural Networks for Signal Processing)*, Martigny, Switzerland, 2002.

[13] A. Hyvärinen and P. Hoyer. "Emergence of Phase and Shift Invariant Features by Decomposition of Natural Images into Independent Feature Subspaces", *Neural Computation*, Vol. 12, pp. 1705-1720, 2000.

[14] I. Jolliffe. *Principle Component Analysis*, 2nd Ed., Springer Series in Statistics, Springer-Verlag, New York, 2002.

[15] D. Lee and H. Seung. "Learning the Parts of Objects by Non-Negative Matrix Factorization", *Nature*, Vol. 401, pp. 788-791, 1999.

[16] D. Lee and H. Seung. "Algorithms for Non-Negative Matrix Factorization", *Advances in Neural Processing*, 2000.

[17] W. Liu and J. Yi. "Existing and New Algorithms for Non-negative Matrix Factorization", preprint, Computer Sciences Department, University of Texas at Austin, 2003.

[18] B. Mclean and P. Elkind. *The Smartest Guys in the Room: The Amazing Rise and Scandalous Fall of Enron*, Portfolio, 2003.

[19] Z. Mu, R. Plemmons and P. Santago. "Iterative Ultrasonic Signal and Image Deconvolution for Estimating the Complex Medium Response", preprint, submitted to *IEEE Transactions on Ultrasonics and Frequency Control*, 2003.

[20] S. Prasad, T. Torgersen, V. Pauca, R. Plemmons, and J. van der Gracht. "Restoring Images with Space Variant Blur via Pupil Phase Engineering", Optics in Info. Systems, Special Issue on Comp. Imaging, SPIE Int. Tech. Group Newsletter, Vol. 14, No. 2, pp. 4-5, 2003.

[21] F. Shanaz, M.W. Berry, V.P. Pauca, and R.J. Plemmons. "Document Clustering Using Nonnegative Matrix Factorization", *Information Processing & Management*, 2005, to appear.

[22] S. Wild, J. Curry and A. Dougherty. "Motivating Non-Negative Matrix Factorizations", *Proceedings of the Eighth SIAM Conference on Applied Linear Algebra*, Williamsburg, VA, July 15-19, 2003. See `http://www.siam.org/meetings/la03/proceedings/`.

[23] W. Xu, X. Liu, and Y. Gong. "Document-Clustering based on Non-Negative Matrix Factorization", *Proceedings of SIGIR'03*, July 28 - August 1, Toronto, CA, pp. 267-273, 2003.

Structure in the Enron Email Dataset

P.S. Keila and D.B. Skillicorn
School of Computing
Queen's University
{keila,skill}@cs.queensu.ca

Abstract

We investigate the structures present in the Enron email dataset using singular value decomposition and semidiscrete decomposition. Using word frequency profiles we show that messages fall into two distinct groups, whose extrema are characterized by short messages and rare words versus long messages and common words. It is surprising that length of message and word use pattern should be related in this way. We also investigate relationships among individuals based on their patterns of word use in email. We show that word use is correlated to function within the organization, as expected. We also show that word use among those involved in alleged criminal activity may be slightly distinctive.

1 Introduction

Many countries intercept communication and analyze messages as an intelligence technique. The largest such system is Echelon [3], run jointly by the U.S., Canada, U.K, Australia, and New Zealand. The standard publicly-acknowledged analysis of intercepted data is to search messages for keywords, discard those messages that do not contain keywords, and pass those that do to analysts for further processing. An interesting question is what else can be learned from such messages; for example, can connections between otherwise innocuous messages reveal links between their senders and/or receivers [13].

The Enron email dataset provides real-world data that is arguably of the same kind as data from Echelon intercepts – a set of messages about a wide range of topics, from a large group of people who do not form a closed set. Further, individuals at Enron were involved in several apparently criminal activities. Hence, like Echelon data, there are probably patterns of unusual communication within the dataset.

Understanding the characteristics and structure of both normal and abnormal (collusive) emails therefore provides information about how such data might be better analyzed in an intelligence setting.

Linguistically, email has been considered to occupy a middle ground between written material, which is typically well-organized, and uses more formal grammatical style and word choices; and speech, which is produced in real-time and characterized by sentence fragments and informal word choices. Although the potential for editing email exists, anecdotal evidence suggests that this rarely happens; on the other hand, email does not usually contain the spoken artifacts of pausing (ums etc.).

We examine the structure of the Enron email dataset, looking for what it can tell us about how email is constructed and used, and also for what it can tell us about how individuals use email to communicate.

2 Related Work

Previous attention has been paid to email with two main goals: spam detection, and email topic classification. Spam detection tends to rely on local properties of email: the use of particular words, and more generally the occurrence of unlikely combinations of words. This has been increasingly unsuccessful, as spam email has increasingly used symbol substitution (readable to humans) which makes most of its content seem not to be words at all.

Email topic classification attempts to assist

users by automatically classifying their email into different folders by topic. Some examples are [2, 7, 10, 12]. This work has been moderately successful when the topics are known in advance, but perform much less adequately in an unsupervised setting (but see some of the papers in this workshop). An attempt to find connections between people based on patterns in their email can be found in [8].

3 Matrix Decompositions

We will use two matrix decompositions, *Singular Value Decomposition* (SVD) [4], and *SemiDiscrete Decomposition* (SDD) [5, 6]. Both decompose a matrix, A, with n rows and m columns into the form

$$A = C\ W\ F$$

where C is $n \times k$, W is a $k \times k$ diagonal matrix whose entries indicate the importance of each dimension, and F is $k \times m$.

There are several useful ways to interpret such a decomposition. The *factor* interpretation regards the k rows of F as representing underlying or latent factors (and hence better explanations of the data) while the rows of C describe how to mix these factors together to get the observed values in A. The *geometric* interpretation regards the k rows of F as representing axes in some transformed space, and the rows of C as coordinates in this (k-dimensional) space. The *layer* interpretation relies on the fact that A is the sum of k outer product matrices, A_i, where each A_i is the product of the ith column of C and the ith row of F (and the ith diagonal element of W). All of these interpretations can be helpful in interpreting a dataset.

Singular value decomposition is usually interpreted using the factor model (in the social sciences) and the geometric model (in the sciences). An SVD for the matrix A is

$$A = U\ S\ V'$$

where U and V are orthonormal, the diagonal of S is non-increasing, and $k \leq m$. The usefulness of SVD comes primarily from the fact that the columns of V are orthogonal and hence represent independent factors, or orthogonal axes. The first k columns of U can be interpreted as the coordinates of a point corresponding to each row of A in a k-dimensional space, and that this is the most faithful representation of the relationships in the original data in this number of dimensions.

The correlation between two objects is proportional to the dot product between their positions regarded as vectors from the origin. Two objects that are highly correlated have a dot product (the cosine of the angle between the two vectors) that is large and positive. Two objects that are highly negatively correlated have a dot product that is large and negative. Two objects that are uncorrelated have dot product close to zero.

This property is useful because there are two ways for a dot product to be close to zero. The obvious way is for the vectors concerned to be orthogonal. However, when m is less than n (as it typically is) there are many fewer directions in which vectors can point orthogonally than there are vectors. Hence if most vectors are uncorrelated, they must still have small dot products but cannot all be orthogonal. The only alternative is that their values must be small. Hence vectors that are largely uncorrelated must have small magnitudes, and the corresponding objects are placed close to the origin in the transformed space. Hence, in a transformed space from an SVD, the points corresponding to objects that are 'uninteresting' (they are correlated either with nothing or with everything) are found close to the origin, while points corresponding to interesting objects are located far from the origin (potentially in different direction indicating different clusters of such objects).

The SemiDiscrete Decomposition (SDD) of a matrix A is

$$A = X\ D\ Y$$

where the entries of X and Y come from the set $\{-1, 0, +1\}$, D is a diagonal matrix, and k can have any value, not necessarily less than m. The natural interpretation of SDD is a layer one [9]. Each A_i corresponds to a column of X and a row of Y, weighted by an entry from D. The product of x_i and y_i is a stencil representing a 'bump' (where the product has a $+1$) and corresponding 'ditch' (where the product has a -1). The corresponding value of D gives the height of the bump and ditch

at each level. Hence an SDD expresses a matrix as the sum of bumps, with the most significant bumps appearing first. Because the choice of the sequence of bumps depends on both their area (how many locations in the matrix they cover) and their height, altering the scale of A will change the resulting SDD. In particular, taking the signed square of each value in the matrix will give greater emphasis to the heights of bumps and hence select outlying regions of the dataset earlier. Conversely, taking the signed square root of each value in the matrix will tend to find large homogeneous regions earlier.

SDD generates a ternary, unsupervised hierarchical classification of the samples, based on the values in each successive column of the X matrix. Consider the first column of X. Those samples for which this column has the value $+1$ can be grouped; those samples for which this column has the value -1 are, in a sense, similar but opposite; and those samples for which this column has the value 0 are unclassified at this level. This can be repeated for columns 2, 3, and so on, to produce a classification tree.

Neither SVD nor SDD exploit the order of rows and columns in the data matrix, so they do not start with any advantage over more conventional data-mining techniques.

4 Structure from Word Usage

Most emails contain few words from the possible vocabulary, so a word-document (word-email) matrix is extremely sparse. Although SVD could be performed on such matrices using sparse matrix techniques such as Lanczos methods, we chose instead to analyze matrices whose rows correspond to emails and whose columns correspond to frequency in the email. The entries in the matrix are the (global) ranks of words in frequency order in the message. For example, if the most frequent words in an email is "stock" and this word ranks 12,000th overall in the Enron noun frequency list, then the entry in the row corresponding to that email and the first column of the matrix is 12,000.

Two emails are similar in this representation if they have similar word usage profiles *in descending order of frequency*; in other words, the similarity metric is more discriminating than one based only on a bag-of-words similarity metric.

Basic Structure An SVD analysis of the entire email dataset is shown in Figure 1, based on 494,833 messages using 160 203 distinct words (no stemming has been applied).

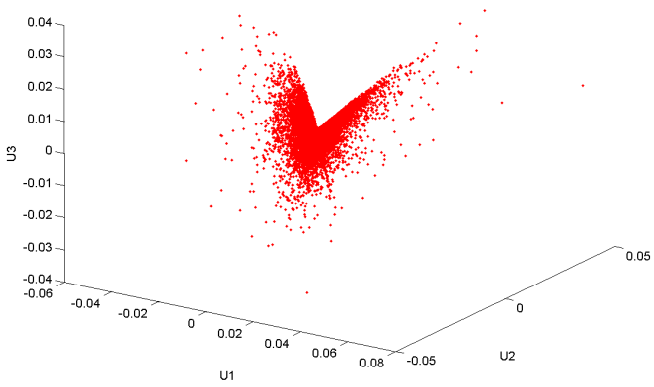


Figure 1: SVD plot of entire email set of 494,833 messages. Note the strong bifurcation.

The most obvious and striking feature of this plot is that it results in a 'butterfly' shape, that is the emails separate into two clusters that grow increasingly different with distance from the origin. This separation is quite surprising; as far as we are aware previous analysis of email datasets has revealed separation by topic, but not such a strong structural separation. This structure remains more or less fixed as the set of nouns is reduced, indicating that it is not an artifact of particular choice of nouns under consideration.

To explore the structure of the dataset more deeply, we reduced the number of words under consideration by removing those we believed made the least contribution to interesting structure. We used the BNC corpus [1], which is a frequency-ranked list of words in both spoken and written English to assist. We first removed words that appear in the Enron dataset but not in the BNC corpus. This removes almost all of the strings that are not real words (artifacts of email processing and also of postprocessing of the dataset); and also almost all of the proper names and acronyms.

We also removed words that were very frequent (appeared more than 1000 times in the dataset) and very infrequent (appeared fewer than 20 times in the dataset). Reducing the set of words removes some emails entirely. Figure 2 shows the SVD plot for this reduced dataset. As expected, the 'less interesting' emails are the ones that disappear, and a secondary structure begins to appear. The two 'wings' reduce to borders, and there are marked extensions that extend into the page on the left wing and out of the page on the right – in other words, the overall shape becomes a spiral.

We reduced the word set further by retaining only words whose frequency of use in the email dataset is greater than their frequency of use in English (as recorded in the BNC corpus). This restricts attention to the 7424 words that Enron people use to communicate amongst themselves more than the general population. We call this *Enronspeak*, the normal patterns of utterance within the organization.

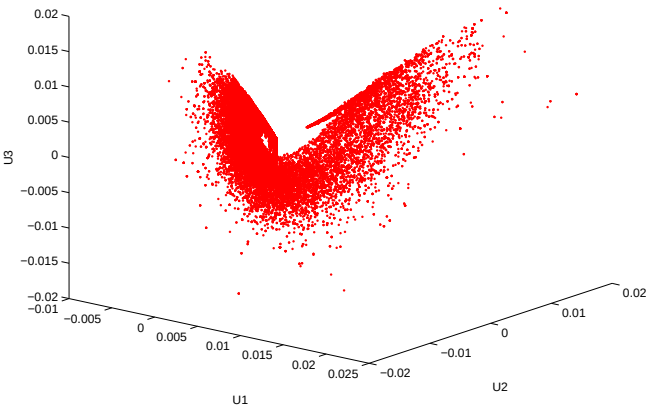


Figure 2: SVD plot of 350,248 emails, when the word set is reduced by (a) removing all words that appear in the Enron emails but not in the BNC corpus, and (b) removing all words with frequency greater than 1000 or less than 20.

This further reduces the number of email messages. An SVD plot is shown in Figure 3. The spiral shape is now very pronounced.

The reason for the strong bifurcation of emails

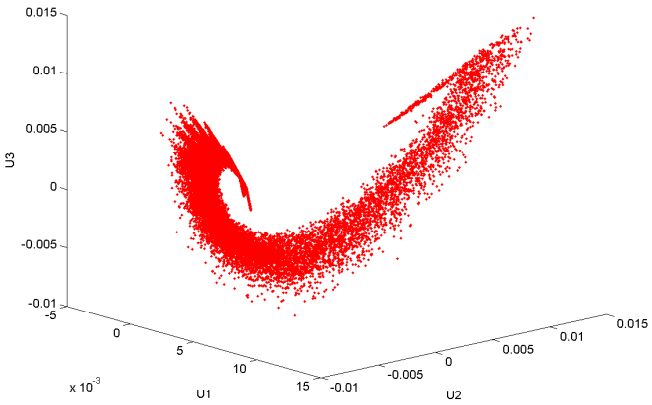


Figure 3: SVD plot of 289,695 emails, when the word set is reduced further by removing words whose frequency is greater in Enron email than in the BNC corpus (Enronspeak) – a set of 7424 words. The left hand goes into the page, while the right hand end comes out of the page.

is not clear. In general, the left hand 'wing' consists of messages with few distinct nouns; the emails near the origin are messages with a moderate number of distinct nouns, and the right hand 'wing' consists of messages with many distinct nouns.

Recall that distance from the origin is a surrogate for interestingness, at least with respect to correlation structure. This spiral shape shows that there are three ways for an email to be uninteresting:

1. It contains very few distinct words (the sharp spike at the back of the left hand wing, which ends up quite near the origin);

2. It is of moderate size and uses words in ordinary ways (the region near the origin);

3. It is very long, and contains so many different nouns that it correlates with many of the other emails (the sharp spike at the front of the right hand wing which also ends up quite near the origin).

The remaining extremal emails are those that have the most interesting correlational structure. Words on the right wing use more nouns altogether,

and so have greater opportunities for interesting correlation, whereas nouns on the left wing use few nouns and so have fewer opportunities. Hence the butterfly structure is quite asymmetric, with the right wing much larger and further from the origin than the left. Figure 4 shows the word frequency profile for a typical extremal message on the left wing. Figure 5 shows the word frequency profile for an extremal message on the right wing.

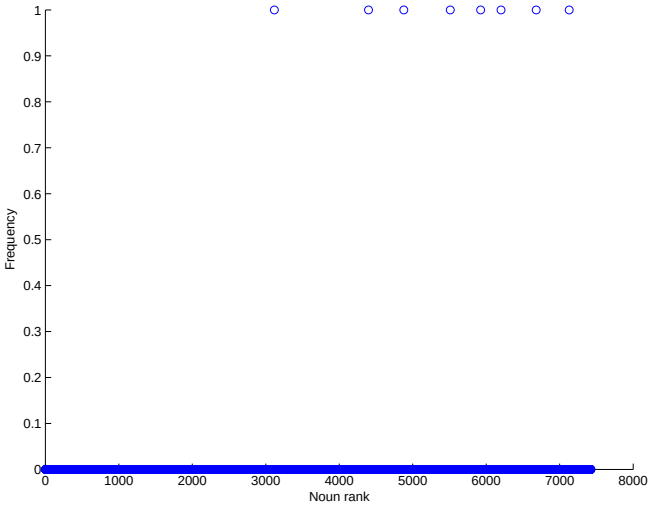


Figure 4: Noun frequency distribution for a typical extremal message on the left wing.

Extremal emails on the left wing can be characterized as: having been composed by a single author, short (in Enronspeak, although potentially containing many ordinary words), and tending to use each noun only once. Extremal emails on the right wing can be characterized as: coming from outside Enron, either digests with many different topics (sports updates, general news) or emails that reference many proper names, long (containing 100-350 Enronspeak nouns), and having more typical word frequency (Zipf-like) profiles.

Figures 6 and 7 show the way in which other properties correlate with position in the SVD plot. Figure 6 shows that message length correlates well with position along the spiral. Figure 7 shows that infrequent words are much more likely to occur at the left hand end, and frequent words to occur at the right hand end. Hence, message length is,

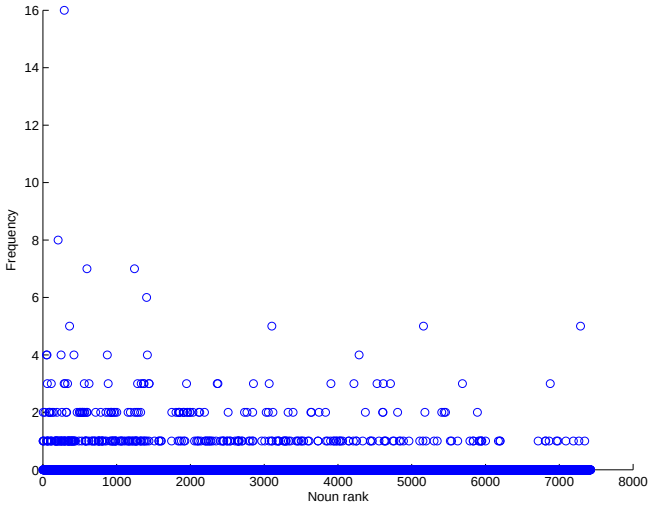


Figure 5: Noun frequency distribution for a typical extremal message on the right wing.

at least to some extent, inversely correlated with rareness of words used.

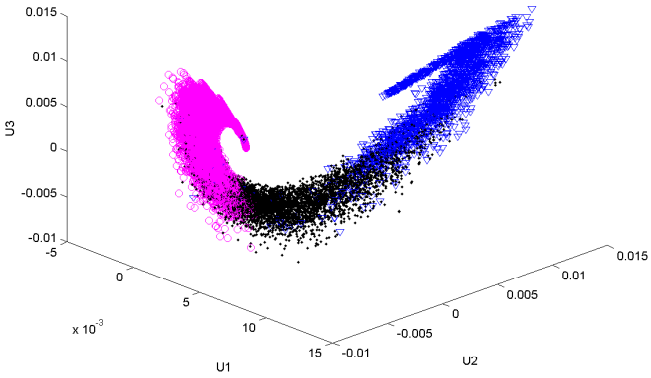


Figure 6: SVD plot labelled by message length (magenta: < 20 nouns; black: < 70 nouns)

Figure 8 shows the relationship between emails and their senders. The Corporate Policy Committee (CPC) consisted of 15 influential executives at Enron. These executives included the CEO, Chairman, Vice-Chairman, CFO, CAO, a number of heads from different Enron divisions, and an in-house lawyer. One member from this com-

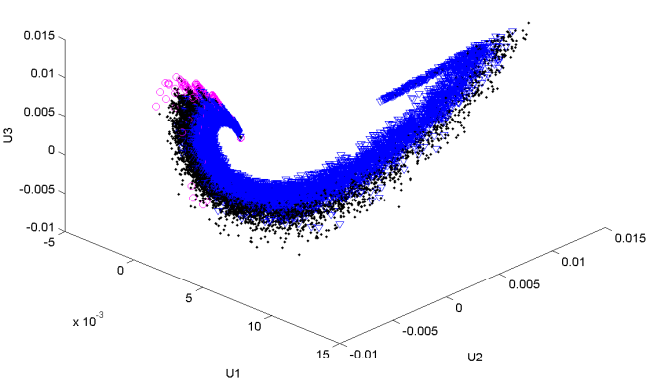


Figure 7: SVD plot labelled by average noun frequency rank (magenta: > 14,000; black: > 8000).

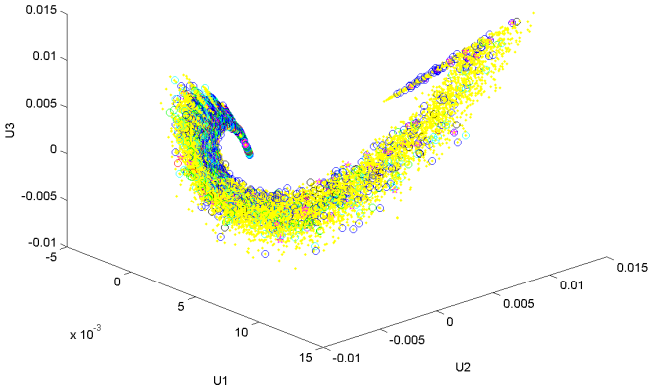


Figure 8: SVD plot labelled by email senders from the CPC. Magenta circle: Delaney; black circle: Derrick; red circle: Horton; blue circle: Kean; green circle: Lay; cyan circle: Skilling; magenta star: Whalley.

mittee has since committed suicide, four have been charged and found guilty of various accounting and securities frauds, and three have been indicted. The figure shows the distribution of emails for those members of the committee whose emails remain in the dataset. Kean was responsible for circulated summaries of references to Enron in the media, and this explains his unusual email profile and relationships.

Figure 9 shows that the interestingness of an email (measured by distance from the origin) peaks for messages with about 220 total nouns, dropping to an asymptote for longer messages. This is surprising, since these messages contain several thousand words.

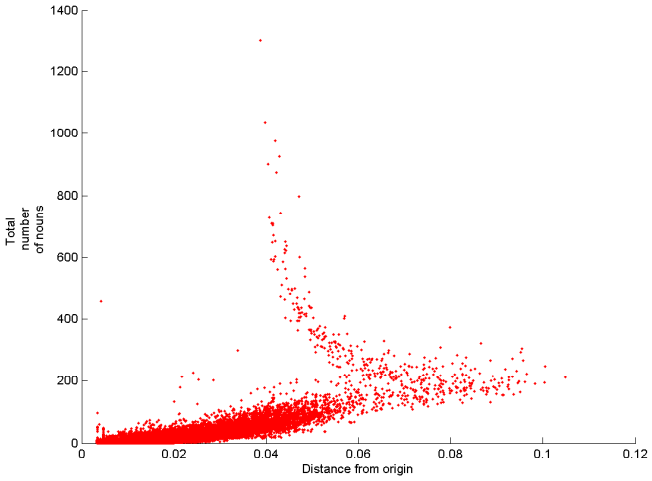


Figure 9: Plot of interest (i.e. distance from the origin in an SVD plot) versus total number of nouns in the message.

5 Authors and Emails

We now consider the matrix whose objects are individuals and whose columns are word frequency, aggregated over all of their emails in the dataset. Hence each row captures a characteristic word use pattern for an individual. More interestingly, correlation in word use patterns determines position in an SVD plot, so that individuals with similar patterns will be placed close together. We might expect that individuals with similar job responsibilities and similar rank might use words in similar ways, both because of writing style, and because of similarity in typical subject matter. Further details of participants and their situation within Enron can be found in [11].

Figure 10 shows an SVD plot with a point for each individual in the dataset. The basic structure

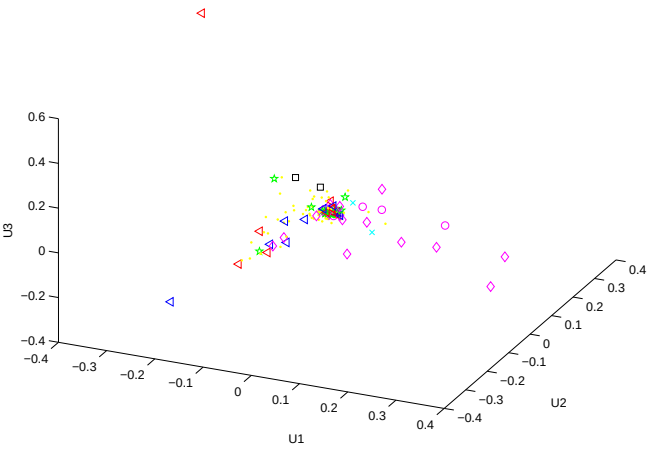


Figure 10: Relationships among 150 individuals based on similarity of email word use. Magenta: VP (diamond), President (circle); Black: CEO; Green: Director; Blue: Trader; Red: Manager; Cyan: Lawyer; Yellow: Unknown/Other. In this and subsequent figures, a set of 1713 words used by no more than 15 people are used.

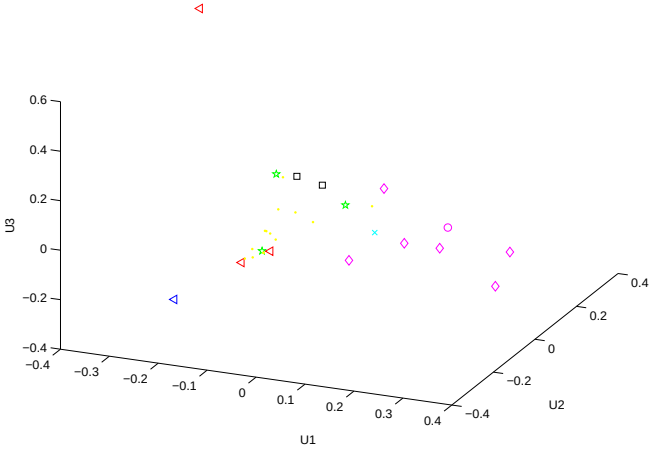


Figure 11: Relationships among 30 most interesting individuals. Labelling as in Figure 10

is a T-shape, with Vice-presidents along one arm towards the bottom right, and traders and other managers towards the bottom left. Core figures in the company tend to appear close to the center.

We can further restrict our attention to the individuals whose distance from the origin in the SVD plot is greater than the median distance. This leaves 30 individuals, including most of those with a significant role in the organization.

Figure 11 shows the SVD plot of the 30 most interesting individuals.

Figure 12 shows the same plot, but with the points labelled by their SDD classification. Note how the (unsupervised) clustering properly distinguishes the functional properties of these individuals. Note also that the SDD labelling agrees, in general, with the positional similarities from SVD.

We can also add weights to certain rows and columns in the raw data. This has the effect of moving them away from the origin, and hence making them seem more important – but it also tends to cause correlated objects or attributes to follow them. We experiment with this by increasing

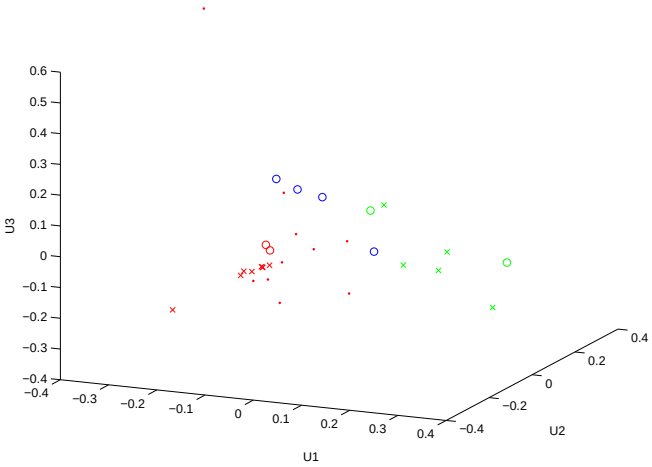


Figure 12: Relationships among 30 most interesting individuals, labelled by SDD classification.

the weight on words used by Lay and Skilling by a factor of 1.4. The result is shown in Figure 13. The effect is to begin to partition the entire set of words into two clusters, one perhaps corresponding to the language of senior executives, and the other to the language of ordinary organization members.

Figure 15 plots the positions of individuals by word use, when the words used by Lay and Skilling are weighted by 1.4. Several other pairs of

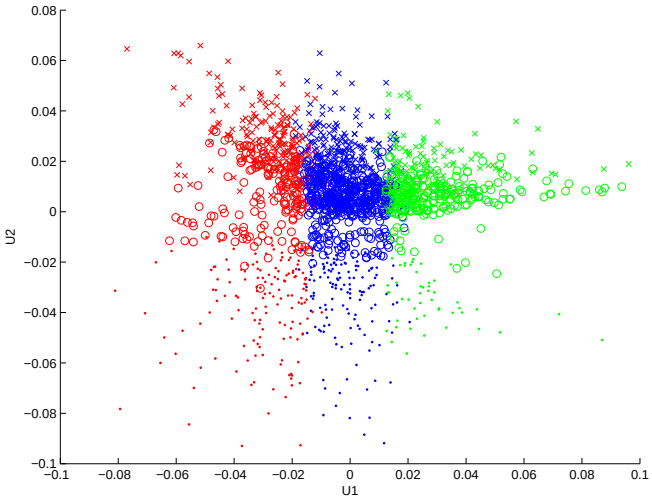


Figure 13: SDD labelled plot of words, weighting emails from Lay and Skilling by 1.4.

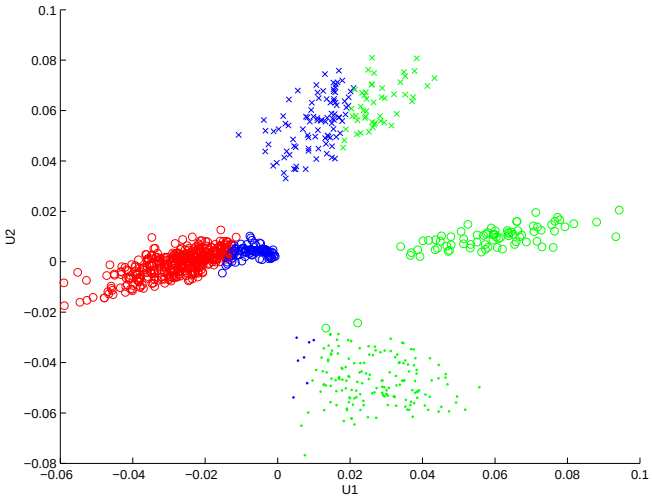


Figure 14: SDD labelled plot of words, weighting emails from Lay and Skilling by 2. The clusters at the top and right are words used disproportionately by Lay and Skilling; The cluster at the left is words that are rare; the cluster at the bottom is words used by individuals on the CPC but not by Lay and Skilling.

individuals move into closer proximity compared to Figure 11. This may reflect particular topics about which these pairs, as well as Lay and Skilling, exchanged emails.

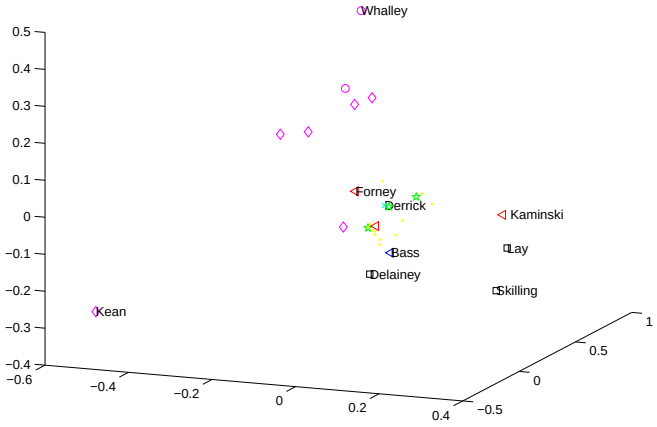


Figure 15: SVD plot of individuals when words used by Lay and Skilling are weighted by 1.4. Lay and Skilling move closer together, but so do Bass and Delainey; and Forney and Derrick.

6 Changes in word usage over time

We divided the email set into four sections covering periods in the years 1999 to 2001. Our first subset is a collection of all the emails sent and received in the year 1999. Enron's first attempt at manipulating energy prices in California occurred in May of 1999. Although reprimanded for the attack, Enron traders engaged in substantially the same conduct the following spring under the schemes Death Star, Ricochet, Fat Boy, and Get Shorty. Hence there is reason to believe that traders at Enron were devising ways to game the newly deregulated energy market in California in the latter half of 1999.

The second subset is the collection of all of the emails sent and received in the year 2000. From May to August of 2000, the West Coast trading desk at Enron booked over $200 million in profits, which is roughly four times the profit the desk had made in all of 1999. It was also the first time since the end of World War II that power companies in California were forced to declare rolling blackouts.

The third and fourth subsets are the emails sent and received in 2001, divided by the time when Jeff

Skilling left the company and it began its public fall.

For each subset of emails, we used the same set of 1713 nouns used above. We then created a noun-usage profile for each user over each of the 4 time periods. The resulting graphs can be seen in Figures 16, 17, 18, and 19.

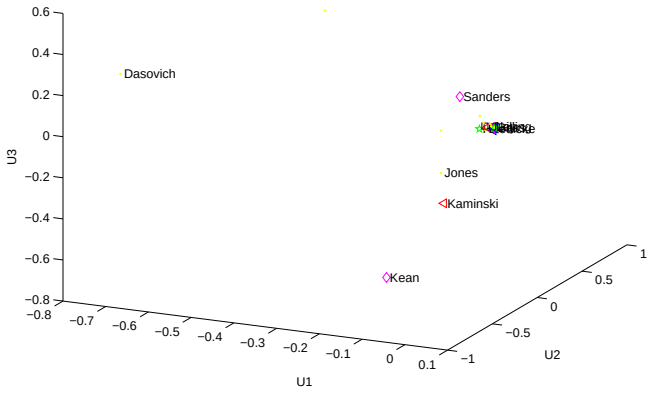


Figure 16: SVD plot of the top 30 most interesting individuals based on word usage – 1999

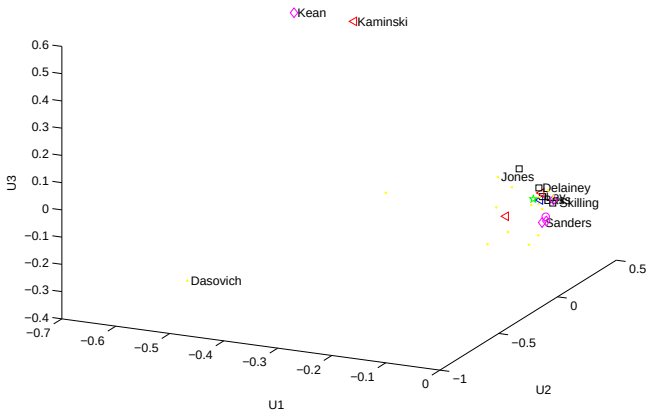


Figure 17: SVD plot of the top 30 most interesting individuals based on word usage – 2000

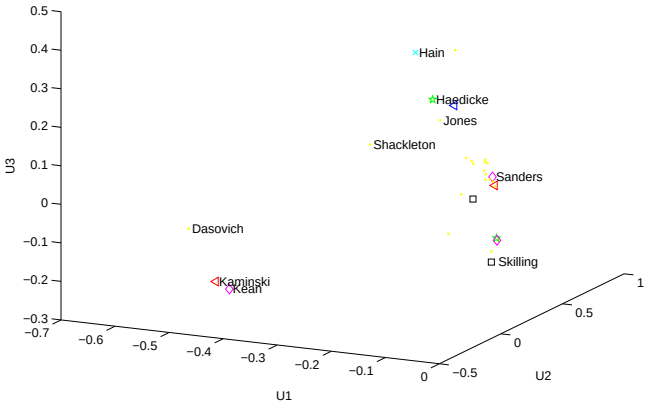


Figure 18: SVD plot of the top 30 most interesting individuals based on word usage – 2001 before Skilling's departure

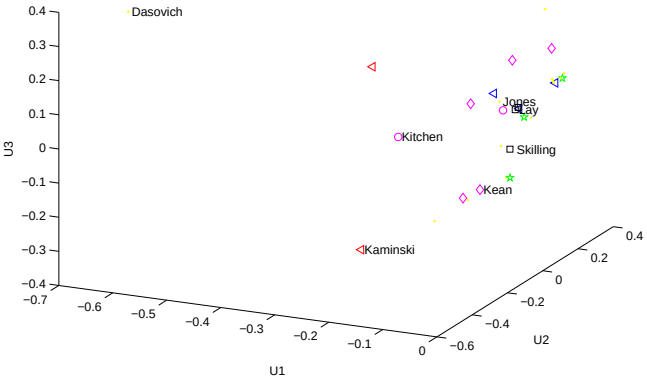


Figure 19: SVD plot of the top 30 most interesting individuals based on word usage – 2001 after Skilling's departure

In each of the four figures, Kaminski, Kean, and Dasovich are far from the origin and hence interesting. Their word use patterns are significantly different from the rest of the company. Knowing the role Kean and Kaminski played in the company,

this is not entirely surprising. Dasovich's appearance, however, is less expected. Dasovich was an Enron government relations executive.

Hain and Haedicke, members of the Enron general counsel, make an appearance in the first half of 2001 (Figure 18). Jones, an unknown character in the Enron saga, maintains a close correlation to Haedicke. In late 2000, Haedicke was made fully aware of the activities of the West Coast trading desk and began to think of ways to protect Enron's role in the affair. Skilling took his first extended vacation in June of 2001 and formally resigned that August.

Knowing that the Enron's trading business came under scrutiny in the later half of 2001 it is not surprising to see that Kitchen, President of Enron Online, appears in Figure 19 and, although similar to Kean, is farther from the origin.

7 Conclusions

Using matrix decompositions such as singular value decomposition and semidiscrete decomposition, we have explored the structure of a large real-world email corpus. The structure of email messages, using similarity based on word use frequency profiles shows a distinctive butterfly/spiral pattern which we have not been able to fully account for. There appears to be a strong differentiation between short messages using rare (in this context) words, and long messages using more typical words. The characteristic length of the emails with the most interesting correlative structure seems surprisingly long.

We also analyzed the relationships among individuals based on the word use frequency profiles of the emails they send. This showed a clear effect of company role on such relationships – individuals of similar status and role tend to communicate in similar ways. There are some hints that emphasizing certain words tends to pull together individuals who are not obviously associated in the company environment, but there may be several explanations for this behavior. It is also clear that word usage patterns are affected by major changes in the company environment, and that this can be used to track changes in relationships among individuals.

References

[1] British National Corpus (BNC), 2004. **www.natcorp.ox.ac.uk**.

[2] W.W. Cohen. Learning to classify English text with ILP methods. In L. De Raedt, editor, *Advances in Inductive Logic Programming*, pages 124–143. IOS Press, 1996.

[3] European Parliament Temporary Committee on the ECHELON Interception System. Final report on the existence of a global system for the interception of private and commercial communications (echelon interception system), 2001.

[4] G.H. Golub and C.F. van Loan. *Matrix Computations*. Johns Hopkins University Press, 3rd edition, 1996.

[5] G. Kolda and D.P. O'Leary. A semi-discrete matrix decomposition for latent semantic indexing in information retrieval. *ACM Transactions on Information Systems*, 16:322–346, 1998.

[6] T.G. Kolda and D.P. O'Leary. Computation and uses of the semidiscrete matrix decomposition. *ACM Transactions on Information Processing*, 1999.

[7] D. Lloyd and N. Spruill. Security screening and knowledge management in the Department of Defense. In *Federal Conference on Statistical Methodology*, 2001.

[8] R. McArthur and P. Bruza. Discovery of implicit and explicit connections between people using email utterance. In *Proceedings of the Eighth European Conference of Computer-supported Cooperative Work, Helsinki*, pages 21–40, 2003.

[9] S. McConnell and D.B. Skillicorn. Semidiscrete decomposition: A bump hunting technique. In *Australasian Data Mining Workshop*, pages 75–82, December 2002.

[10] C. OBrien and C. Vogel. Exploring the subject of email filtering: Feature selection in statistical filtering, submitted, 2004.

[11] J. Shetty and J. Adibi. The Enron email dataset database schema and brief statistical report. Technical report, Information Sciences Institute, 2004.

[12] A.F. Simon and M. Xenos. Dimensional reduction of word-frequency data as a substitute for intersubjective content analysis. *Political Analysis*, 12:63–75, 2004.

[13] D.B. Skillicorn. Detecting related message traffic. In *Workshop on Link Analysis, Security and Counterterrorism, SIAM Data Mining Conference*, pages 39–48, 2004.

Live and Dead Nodes

S. Lehmann*

Abstract

In this paper, we explore the consequences of a distinction between 'live' and 'dead' network nodes; 'live' nodes are able to acquire new links whereas 'dead' nodes are static. Based on this distinction, we develop an analytically soluble growing network model incorporating this feature; and show how well this model corresponds to the empirical network constituted by citations and references (in- and out-links) between papers (nodes) in the SPIRES database of scientific papers in high energy physics. We also demonstrate that the death mechanism alone can result in power law degree distributions for the resulting network.

1 Introduction

The study and modeling of complex networks has expanded rapidly in this new millennium and is now firmly established as a science in its own right [1, 2, 3, 4]. One of the oldest examples of a large complex network is the network of citations and references (in- and out-links) between scientific papers (nodes) [5, 6, 7, 8, 9]. A very successful model describing networks with power-law degree distributions is based on what we shall call *preferential attachment*. The principles underlying this model were first introduced by Simon [10], applied to citation networks by de Solla Price [11][1], and independently rediscovered by Barabási and Albert [12]. Various modifications of the preferential attachment model have assumed a prominent position in the modeling literature following Barabási's rediscovery in 1999; in the current context, the key papers on preferential attachment are [7, 8, 13, 14, 15]. The primary strength of the preferential attachment model is its simplicity but simultaneously, this strength is also the model's weakness. For example, preferential attachment models tend to assume that networks are homogeneous, but when a network can be shown to have significant and identifiable inhomogeneities (this is the case for the citation network), the data can compel us to augment the preferential attachment model in order to account for this feature.

The primary conclusion of Ref. [7] is that the majority of nodes in a citation network 'die' after a short time never to be cited again, whereas, a small population of papers remains 'alive' and continues to be cited many years after publication. In Ref. [8] it was established that this distinction between alive and dead papers is an important inhomogeneity in the citation network that is not accounted for by the simple preferential attachment model. Interestingly, a similar distinction between alive and dead nodes was recently independently suggested by [9]. In this paper, we will explore how the distinction between live and dead papers manifests itself in our network models and thus, suggest an extension of the preferential attachment model.

2 The SPIRES data

The work in this paper is based on data obtained from the SPIRES[2] database of papers in high energy physics. More specifically, our dataset is the network of all citable papers from the theory subfield, ultimo October 2003. After filtering out all papers for which no information of time of publication is available and removing all references to papers not in SPIRES, a final network of 275 665 nodes and 3 434 175 edges remains.

Above we described a dead node as one that no longer receives citations, but how does one go about defining a dead node in *real* data? We have tested several definitions, and the results are qualitatively independent of which specific definition is chosen. Therefore, we can simply define live papers as papers cited in 2003. We acknowledge that there are papers that receive citations after a long period of being dormant, but these cases are rare and do not affect the large scale statistics. In Figure 2, the (normalized) degree distributions of live and dead papers in the SPIRES data are plotted, and we can clearly see that the two distributions differ significantly. Having isolated the dead papers, we

*Informatics and Mathematical Modeling, Technical University of Denmark, DK-2800 Kgs. Lyngby, Denmark.

[1] More precisely, de Solla Price was the first person to re-think Simon's model and use it as a basis of description for *any* kind of network, cf. [4].

[2] SPIRES is an acronym for 'Stanford Physics Information REtrieval System' and is the oldest computerized database in the world. The SPIRES staff has been cataloguing all significant papers in high energy physics and their lists of references since 1974. The database is open to the public and can be found at http://www.slac.stanford.edu/spires/.

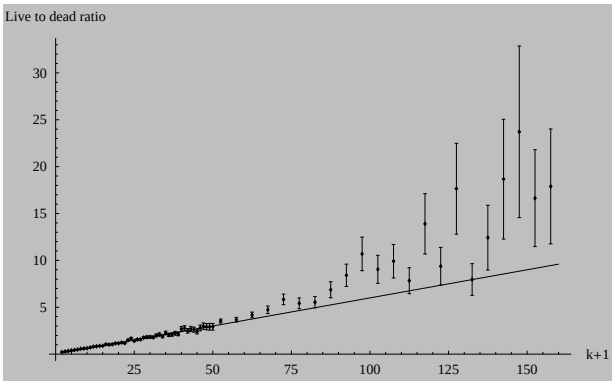


Figure 1: Displayed above is ratio of live to dead papers as a function of k. Error bars are calculated from square roots of the citation counts in each bin. Also, a straight line is present to illustrate the linear relationship between the live and dead populations for low values of k.

are not only able to plot them; we can also determine the empirical ratio of live to dead as a function of the number of citations per paper, k. In Figure 1 this ratio is displayed with k ranging from 1 to 150 (papers with zero citations are dead by definition). Over most of this range, the data is well described by a straight line. Note that the data for dead papers with high citation counts is very sparse. For example, only 0.15% of the dead papers have more than 100 citations, so the statistics beyond this point are highly unreliable. More generally, plotting the ratio of live to dead papers in a linear representation is a very pessimistic representation of the data. We therefore conclude that the ratio of *dead to live* papers is relatively well described by the simple form $1/(k+1)$ for all except the highest values of k, where the number of dead papers is overestimated by a factor of two to three. In the following section, we will make use of this relation to extend the preferential attachment model to include dead nodes.

3 The Model

The basic elements of the preferential attachment model are *growth* and *preferential attachment* [12]. The simplest model starts out with a number of initial nodes and at each update, a new node is added to the database. Each new node has m out-links that connect to the nodes already in the database. Each new node enters with $k = 0$ real in-links. This part is the *growth* element of the model. Note that, since we have chosen to eliminate all references to papers not in SPIRES from the dataset, there is a sum rule that the average number of citations per paper is also m. The *preferential attachment* enters the model because we assume the probability for each node already in the database, to receive one of the m new in-links to be proportional to its current number of in-links. In order for the newest nodes (with $k = 0$ in-links) to be able to begin attracting new citations, we load each node into the database with $k_0 = 1$ 'ghost' in-links that can be subtracted after running the model; we then let the probability of acquiring new citations be proportional to the *total* number of in-links, both real and ghost in-links.

One of the simplest ways to augment the simple incarnation of the preferential attachment model described above is to regard k_0 as a free parameter. This allows us to estimate when the effects of preferential attachment become important. Since there is no *a priori* reason why a paper with 2 citations (in-links) should have a significant advantage over a paper with 1 citation, it is preferable to let the data decide. Thus, in our model, the probability that a live paper with k citations acquires a new citation at each time step is proportional to $k + k_0$ with $k_0 > 0$. Also, note that we can think of the displacement k_0 as a way to interpolate between full preferential attachment ($k_0 = 1$) and no preferential attachment ($k_0 \to \infty$).

The significant augmentation of the simple model in this context, however, is that in our model *each paper has some probability of dying at every time step.* From Section 2, we have a very good idea of what this probability should be chosen to be: Figure 1 shows us that for a paper with k citations, this probability is proportional to $1/(k+1)$ to a reasonable approximation. With this qualitative description of the model in hand, let us proceed and solve it.

4 Rate Equations

One very powerful method for solving preferential attachment type network models is the rate equation approach, introduced in the context of networks by [13]. Let L_k and D_k be the respective probabilities of finding a live or a dead paper with k real citations. As explained above, we load each paper into the database with $k = 0$ real citations and m references. The rate equations become

$$\begin{aligned} L_k &= m(\lambda_{k-1}L_{k-1} - \lambda_k L_k) \\ &- \eta_k L_k + \delta_{k,0} \qquad (4.1) \\ D_k &= \eta_k L_k, \qquad (4.2) \end{aligned}$$

where the λ_k and η_k are rate constants. Since every paper has a finite number of citations, the probabilities L_k and D_k become exactly zero for sufficiently large k; we also define L_k to be zero for $k < 0$. In this way, all sums can run from $k = 0$ to infinity. These equations

trivially satisfy the normalization condition

$$\sum_k (L_k + D_k) = 1, \tag{4.3}$$

for any choice of η_k and λ_k. However, we also demand that the mean number of references must equal the mean number of papers

$$\sum_k k(L_k + D_k) = m. \tag{4.4}$$

This constraint must be imposed by an overall scaling of η_k and λ_k. The model described in Section 3 corresponds to a choice of η_k and λ_k, where

$$m\lambda_k = a(k + k_0) \tag{4.5}$$

is the preferential attachment term and

$$\eta_k = \frac{b}{k+1} \tag{4.6}$$

corresponds to the previously described death mechanism. We insert Equations (4.5) and (4.6) into Equation (4.1) and perform the recursion to find

$$\begin{aligned} L_k &= \frac{\Gamma(k+2)}{ak_1k_2} \\ &\times \frac{\Gamma(k+k_0)}{\Gamma(k_0)} \\ &\times \frac{\Gamma(1-k_1)}{\Gamma(k-k_1+1)} \\ &\times \frac{\Gamma(1-k_2)}{\Gamma(k-k_2+1)}, \end{aligned} \tag{4.7}$$

and of course $D_k = bL_k/(k+1)$. The two new constants, k_1 and k_2 are solutions to the quadratic equation

$$(a(k + k_0) + 1)(k + 1) + b = 0 \tag{4.8}$$

as a function of k.

5 The $k_0 \to \infty$ Limit

Before moving on, let us explore the limit where $k_0 \to \infty$ and preferential attachment is turned off. In this regime, the network is, of course, completely dominated by the death mechanism. We can either obtain this limit by going back and solving Equations (4.1) and (4.2) with $\lambda_k = constant$ and $\eta_k = b/(k + 1)$, or we can make the more elegant replacement $\alpha = ak_0$ in Equation (4.7), and then take the limit $k_0 \to \infty$ for fixed α. The two approaches are equivalent. We find

$$L_k = \frac{1}{\alpha}\left(\frac{\alpha}{1+\alpha}\right)^{k+1} \frac{(\frac{b}{1+\alpha})!(k+1)!}{(\frac{b}{1+\alpha}+k+1)!}, \tag{5.9}$$

and the D_k are still simply $bL_k/(k + 1)$. With this expression for L_k, let us investigate what happens in the limit of $\alpha \to \infty$ and $b \to \infty$ with the ratio $r = b/(\alpha + 1) \approx b/\alpha$ fixed. In this limit, it is alluring to replace the term $\alpha/(\alpha+1)$ by one[3]. In this case, the use of identities, such as

$$\sum_{k=1}^{\infty} \frac{k!}{(k+r)!} = \frac{1}{(1-r)r!} \tag{5.10}$$

enable us to compute the fraction of dead papers f, and the average numbers of citations for live and dead papers. The results are simply

$$1 - f = \frac{1}{\alpha - 1} \tag{5.11}$$

$$m_L = \frac{2}{r-2} \tag{5.12}$$

$$m_D = \frac{1}{r-1}, \tag{5.13}$$

and the average number of citations for all papers is evidently $m = (1 - f)m_L + fm_D$. The fraction of dead papers is $f \to 1 - \mathcal{O}(1/b)$ and the average number of citations for all papers approaches m_D.

The most important result, however, is that in this limit we find that

$$L_k \sim \frac{1}{k^r} \quad \text{and} \quad D_k \sim \frac{b}{k^{r+1}}, \tag{5.14}$$

where we assume that $k > r$. Thus, *we see that power law distributions for both live and dead papers emerge naturally in the limit of* $f \to 1$. In the literature, power laws in the degree distributions of networks are often regarded as an indication that preferential attachment has played an essential part in the generation of the network in question; therefore, it is highly interesting to see an alternative and quite different way of obtaining them.

6 The Full Model

Let us now return to the full model and see how it compares to the data from SPIRES. With all zero cited papers in the dead category, the data yields the following average values: $m_L = 34.1$, $m_D = 4.5$ and $m = 12.8$. The fraction of live papers is $f = 27.0\%$. With an rms. error of only 21%, we can do a least squares fit of L_k to the distribution of live papers with parameters $k_0 = 65.6$, $a = 0.436$, and $b = 12.4$. Although only the live data (the squares in Figure 2) is fitted, the agreement with the empirical data in

[3] For present purposes, this is appropriate when $r \geq 2$. When $r < 2$, the neglected factor is essential for ensuring the convergence of the average number of citations for the live and dead papers m_L and m_D.

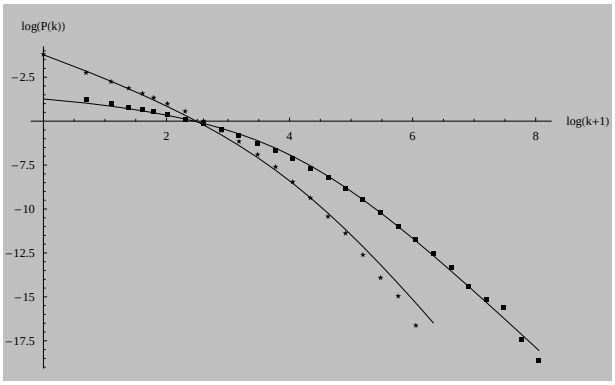


Figure 2: Log-log plots of the normalized degree distributions of live and dead papers. The filled squares represent the live data and the stars represent the dead data. Both lines are the result of a fit to the live data (filled squares) alone.

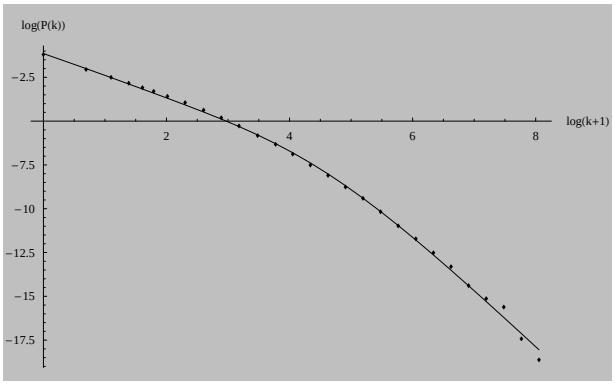


Figure 3: A log-log plot of the normalized degree distribution of all papers (live plus dead). The points are the data; the fit (solid line) is derived from the fit to the live papers (filled squares) in Figure 2.

Figures 2 and 3 is quite striking.

From the model parameters k_0, a, b, we can calculate mean citation numbers for the fit of 32.9, 4.25, and 12.8 for the live, dead, and total population respectively. More interestingly, we learn from the fit that 7.5% of the papers with 0 citations *are actually alive*. If we assign this fraction of the zero-cited papers to the live population, we find the following corrected values for the average values 31.5, 4.6 and 12.5 for the live, dead, and total population respectively. Again, this is a striking agreement with the data. There is so little strain in the fit that we could have determined the model parameters from the empirical values of m_L, m_D, and f. Doing this yields only small changes in the model parameters and results in a description of comparable quality!

Figure 2 reveals that fitting to the live distributions, results in systematic errors for high values of k when we extend the fit to describe the dead papers, but this is not surprising. Recall the similarly systematic deviations from the straight line seen in Figure 1. This figure also explains why the fit to the total distribution shows no deviations from the fit for high k-values even though the total fit includes both live and dead papers—live papers dominate the total distribution in this regime. The obvious way to fix this problem is via a small modification of the η_k. In summary, the full model is able to fit the distributions of both live and dead papers with remarkable accuracy.

One drawback, with regard to the full solution is the relatively impenetrable expression for L_k in Equation (4.7)—associating any kind of intuition to the conglomerate of gamma-functions presented there is difficult. Let us therefore show how L_k can be well approximated by a two power law structure. We begin by noting that, in the limit of large k_0 (as it is the case here), the values of k_1 and k_2 are simply

$$k_1 = -\frac{1}{a} + \frac{b}{ak_0} - k0 \tag{6.15}$$

$$k_2 = -1 - \frac{b}{ak0}. \tag{6.16}$$

Now, let us write out only the k-dependent terms in Equation (4.7) and assign the remaining terms to a constant, C

$$L_k = C\frac{(k+k_0-1)!}{(k-k_1)!}\frac{(k+1)!}{(k-k_2)!} \tag{6.17}$$

$$\approx C\frac{1}{(k+k_0-1)^{1-k_0-k_1}} \times \frac{1}{(k+1)^{-(1+k_2)}} \tag{6.18}$$

$$\approx C\frac{1}{(k+k_0-1)^{1+\frac{1}{a}-\frac{b}{ak_0}}} \times \frac{1}{(k+1)^{\frac{b}{ak_0}}}, \tag{6.19}$$

In Equation (6.18), we have utilized the fact that

$$\frac{(x+s)!}{x!} \approx x^s \tag{6.20}$$

when $x \to \infty$, and in Equation (6.19) we have inserted the asymptotic forms of k_1 and k_2, from Equations (6.15) and (6.16).

This expression for L_k in Equation (6.19) is only valid for large k and k_0, but it proves remarkably accurate even for smaller values of k. With the asymptotic forms of k_1 and k_2 inserted, we can explicitly see that the first power law is largely due to preferential attachment and that the second power law is exclusively due

to the death mechanism. The form for very large k is unaltered by the parameter b. This is not surprising, since there is a low probability for highly cited papers to die. We see that the primary role of the death mechanism in the full model is to add a little extra structure to the L_k for small k.

7 Conclusions

Compelled by a significant inhomogeneity in the data, we have created a model that provides an excellent description of the SPIRES database. It is obvious that the death mechanism ($b \neq 0$) is essential for describing the live and dead populations separately, but less clear that it is indispensable when it comes to the total data. Fitting the total distribution with a preferential attachment only model ($b = 0$) results in $a = 0.528$ and $k_0 = 13.22$ and with a rms. fractional error of 33.6%. This fit displays systematic deviations from the data, but considering that the fit ignores important correlations in the dataset, the overall quality is rather high. The important lesson to learn from the work in this paper, is that even a high quality fit to the global network distributions is not necessarily an indication of the absence of additional correlations in the data.

The most significant difference between the full live-dead model and the model described above is expressed in the value of the parameter k_0. The value of this parameter changes by a factor of approximately 5, from 65.6 to 13.2. Because we find it highly probable that preferential attachment is unimportant until a paper is sufficiently visible for authors to cite it without reading it, we believe that $k_0 \approx 66$ is a much more intuitively appealing value for the onset of preferential attachment. Independent, however, of which value of the k_0 parameter one tends to prefer, the comparison of these two models clearly demonstrates the danger of assigning physical meaning to even the most physically motivated parameters if a network contains unidentified correlations of if known correlations are neglected in the modeling process. Specifically, it would be ill advised to make tenacious conclusions about the onset of preferential attachment if the death mechanism is not included in the model making.

In summary, the live and dead papers in the SPIRES database constitute distributions with significantly different statistical properties. We have constructed a model which includes modified preferential attachment and the death of nodes. This model is quantitatively very successful in describing the found differences. The resulting model has also been shown to produce a two power law structure. The reader should note that this structure provides a beautiful link to the work in [7], where a two power law structure was used to characterize the form of the SPIRES data for purely descriptive reasons without any theoretical foundation. Finally, it has been shown that in the absence of preferential attachment, the death mechanism alone can result in power laws, when the dead nodes dominate the network. Since many real world networks have a large number of inactive nodes and only a small fraction of active nodes, we are confident that this mechanism will find more general use.

Acknowledgements

The author would especially like to thank Andy Jackson without whose generous help and advice this paper would not be possible. Also, thanks to Travis C. Brooks from SPIRES who has supplied the raw data.

References

[1] D. J. Watts. *Small Worlds*. Princeton University Press, Princeton, 1999.

[2] R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74:47–97, 2002.

[3] S. N. Dorogovtsev and J. F. F. Mendes. Evolution of networks. *Advances in Physics*, 51:1079–1187, 2002.

[4] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

[5] D. de Solla Price. Networks of scientific papers. *Science*, 149:510–515, 1965.

[6] S. Redner. How popular is your paper? An emperical study of the citation distribution. *European Physics Journal B*, 4:131–4, 1998.

[7] S. Lehmann, A. D. Jackson, and B. E. Lautrup. Citation networks in high energy physics. *Physical Review E*, 68, 2003.

[8] S. Lehmann , A. D. Jackson and B. E. Lautrup. Life, death, and preferential attachment. *Europhysics Letters*, 2005. Accepted for publication.

[9] S. Redner. Citation statistics from more than a century of physical review. *physics/0407137*, 2004.

[10] H. A. Simon. *Models of Man*. Wiley, New York, 1957.

[11] D. de Solla Price. A general theory of bibliometric and other cumulative advantage processes. *Journal of the American Society for Information Science*, 27:292–306, 1976.

[12] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–12, 1999.

[13] P. L. Krapivsky, S. Redner, and F. Leyvraz. Connectivity of growing random networks. *Physical Review Letters*, 85(21):4629–32, 2000.

[14] P. L. Krapivsky and S. Redner. Organization of growing random networks. *Physical Review E*, 63:066123, 2001.

[15] K. Klemm and V. M. Eguíluz. Highly clustered scale-free networks. *Physical Review E*, 65:036123, 2002.

A Secure Online Algorithm for Link Analysis on Weighted Graph*

Yitao Duan Jingtao Wang Matthew Kam John Canny

Computer Science Division
University of California, Berkeley, CA 94720
{duan, jingtaow, mattkam, jfc}@EECS.Berkeley.EDU

Abstract

Link analysis algorithms have been used successfully on hyperlinked data to identify authoritative documents and retrieve other information. However, existing link analysis algorithms such as HITS suffer two major limitations: (1) they only work in environments with explicit hyperlinked structure such as www, and (2) they fail to capture the rich information that is encoded by patterns of user access or the implicit structure defined by user communication. In this paper we propose the use of weighted graph that is generated and updated via analysis of user behavior to address both issues. We present a generalized HITS algorithm that is suitable for such an approach. The algorithm uses the idea of "lazy update" to amortize cost across a number of updates while still providing accurate ranking to users in real-time. We prove the convergence of the new online algorithm and evaluate its benefit using the Enron email dataset. Finally we devise a scheme that makes the algorithm distributed and privacy preserving using cryptographic techniques thus making it really acceptable in settings such as collaborative work and online community.

1 Introduction

Link analysis algorithms have been used successfully on hyperlinked data to identify authoritative documents and retrieve other information. For instance, the expertise location problem [21, 10, 11, 13] is to find a person in an organization or community who is knowledgeable (and authoritative) in an area. Several approaches [21, 10, 11] construct an explicit social network between individuals, based on email or similar logs, and then use graphical analysis to locate the relevant experts. Similarly, the document ranking problem is to determine the relative levels of "authoritativeness" among a collection of documents. Link analysis algorithms have been used in these environments to uncover such information [12, 1].

*This material is based upon work supported by the National Science Foundation under Grant No. 0222745.

The primary drawback to the above approaches is the need for explicit structure about the social relationships between individuals and the hyperlinks among documents, which do not necessarily exist. For instance, in a computer-mediated environment, a group of individuals could be using tools like software applications to access documents collaboratively, and there is neither an explicit social network representing how each individual is related to others, nor hyperlinks among documents. However, in such context, there are still compelling needs in identifying domain experts and authoritative documents.

Another inadequacy of such algorithms, as Kleinberg acknowledged [12], is that they only make use of the structural information about the graph as defined by the links, and fail to capture patterns of user access which encode essential information about the user's attitude toward the document. The intuition behind the link analysis algorithms is that the link structure encodes important information about the nodes. For example, according to [12], the links among documents, be it hyperlinks on www or citations among academic papers, are constructed consciously by the authors of the documents and represent the authors' "endorsement" toward the authority of documents pointed to by the links and the HITS algorithm [12] can uncover such information to produce a ranking of documents according their level of authority. We believe that a similar principle also holds with patterns of user access: the way a user accesses a document could reflect his/her opinion about it. Meanwhile, a user's level of expertise can also be reflected by the documents that he/she accesses. There is a mutually reinforcing relationship between these two measures, which maps naturally to what Kleinberg denotes "hub" and "authority" [12]: a person is more likely to be an expert in an area if he/she reads more authoritative documents and a document is more likely to be authoritative if it is read by many experts. This phenomenon can also be observed in other graphs such as social networks where the structure is

implicitly defined by communication.

In this paper we propose an approach to address both limitations simultaneously and describe an algorithm that is suitable for such purpose. Notice that access pattern and link structure are not mutually exclusive types of information. Rather, access pattern can complement or even define the other. Our approach uses weighted graphs to model the relationship among nodes and the weights can encode user access or communication. In situations where no explicit link structure exists, these weights effectively *define* the graphical structure and link analysis algorithms can be applied. Where there is an explicit link structure, weights obtained from access or communication analysis can be used to *augment* existing graph and uncover more information.

Using weights in identifying authoritative documents is not new [5]. The novelty of this paper is that we propose the use of weights to model user behavior and construct the link structure. This enables us to apply link analysis algorithms in settings where no such structure exists. However, computing on patterns of access or communication has two implications: (1) instead of a static system, the graph becomes dynamic. The model changes as more data is observed; and (2) user privacy becomes an issue due to the sensitive nature of the user's information used to construct and update the graph. (1) may also mandate that the system services users' query in real-time as there is no end to the accumulation of observations.

To address these new issues, we devised Secure OnlineHITS, a distributed version and enhancement of Kleinberg HITS algorithm that (1) amortizes cost across a number of updates by using "lazy updates", which makes it more suitable for dynamic environments; and (2) uses cryptographic techniques to preserve user's privacy while performing the computation. To make it concrete, we describe the algorithm in the context of document and expertise ranking. However, it is general enough to be applied to other situations where link analysis is appropriate. We use the term document in a broad sense. It refers to any information that can be identified, accessed and analyzed as a unit. For example, a web page or an image can all be classified as a document.

In the rest of the paper, we first review the original HITS algorithm in Section 2. We then discuss the construction of a graph by modelling of user behavior using a weight function in Section 3. In Section 4 we derive an online version of the HITS algorithm to make it more efficient to run in a dynamic environment on accumulated data. Evaluations are presented in Section 5. Finally in Section 6 we discuss privacy and security issues in running such kind of user activity analysis and describe our privacy-enhanced algorithm based on public-key encryption.

2 A Review of HITS

Kleinberg's HITS algorithm [12] is a well-known link analysis algorithm that identifies "authoritative" or "influential" webpages in a hyperlinked environment. Intuitively, by thinking of a hyperlink as a citation, a webpage i is more of an authority (i.e. highly-referenced page) as compared to webpage j if there are more hyperlinks entering i from hub webpages, where a hub is simply a webpage that is a valuable source of links to other webpages. Likewise, a webpage i is a better hub than webpage j if there are more hyperlinks exiting i into authoritative webpages. Given a set of n webpages, HITS first constructs the corresponding n-by-n adjacency matrix A, such that the element in row i and column j of A is 1 if there exists a hyperlink from webpage i to webpage j, 0 otherwise. HITS then iterates the following equations:

$$x^{(t+1)} = A^T y^{(t)} = (A^T A) x^{(t)} \tag{2.1}$$

$$y^{(t+1)} = A x^{(t+1)} = (A A^T) y^{(t)} \tag{2.2}$$

Where the i-th element of x denotes the authoritativeness of webpage i and the i-th element of y denotes the value of webpage i as a hub. With the vectors x and y initialized as vectors of ones and renormalized to unit length at every iteration, as t approaches infinity, $x^{(t+1)}$ and $y^{(t+1)}$ approach x^* and y^*, the principal eigenvectors of $A^T A$ and AA^T, respectively.

Even though HITS is originally intended to locate hubs and authorities in a hyperlinked environment, we observe that hubs and authorities map very well to the users and documents in access based link analysis and the relationship of mutual reinforcement still holds as mentioned in Section 1.

3 Constructing A Weighted Graph

By observing users behavior we can construct a graph of users/documents in environments where no such structure exists. We assume we can observe users's document access and communication pattern using tools like client side logger. Such tools are available from a number of sources. In particular, we have developed a version of our own that runs on Windows platform. Of course such tools have serious privacy implications and we will address such issue in Section 6.

The system consciously logs the users' activities as tuples of the form $< i, j >$, which denotes the fact that user i accesses document j or communicates with user j, depending on the context. These log entries

represent tacit data about the collaborative context because they do not directly encode the links between users nor documents. Given this activity log, we can construct a graph, such that vertices represent the users and/or documents and an edge (i, j) exists and has non-negative weight $w_{i,j}$ iff an item $< i, j >$ exists in the activity log.

How the weight $w_{i,j}$ is computed depends on the application and the goal of the link analysis. The ideas such as TFIDF [5], and the power law of practice [15], etc, are all good heuristics. In some situations the weight can be reduced to simple access or message count. This decision is orthogonal to our work and won't be pursued in this paper. The only assumption we make here is that $w_{i,j}$ is a non-negative, real number.

3.1 Convergence of Weighted HITS Suppose we replace the 0-1 valued element A_{ij} in the adjacency matrix A with a non-negative weight function $w(i, j)$. First we introduce the following two lemmas from [14].

LEMMA 3.1. *If M is a symmetric matrix and v is a vector not orthogonal to the principal eigenvector λ_1 of M, then the unit vector in the direction of $M^k V$ converges to λ_1 as k increases without bound.*

LEMMA 3.2. *If a symmetric M has only non-negative elements, the principal eigenvector of M has only non-negative entries.*

According to the definition of $w(i, j)$, it's easy to see that matrix A has only non-negative values and the symmetric matrix $A^T A$ and AA^T have only non-negative values, thus the principal eigenvectors of $A^T A$ and AA^T have only non-negative entries (lemma 3.2).

If we use a non-negative values vector x, since x is not orthogonal to the eigenvector of AA^T which has only non-negative entries, the sequence $\{y_k\}$ converges to a limit y^* (lemma 3.1). Similarly we can prove that the sequence $\{x_k\}$ converges to a limit x^*.

4 Online HITS

Access based graph construction and link analysis introduces a number of issues of its own such as frequent update, distributed data sources, data security and user privacy concerns, etc. An algorithm alone cannot address all these issues. But a properly designed algorithm can make addressing them a lot easier. In this section we describe a link analysis algorithm that works incrementally as data is being added. We use the idea of "lazy update" to avoid updating and running of the expensive computation so that we can amortize the cost across a number of updates while still maintaining enough precision.

4.1 Basic Approach As shown in Section 1 and 3, the intuition behind HITS fits very well to our application. However, the algorithm is too expensive to run on every update, which can be quite frequent. Recall that the rankings we are seeking, x and y, correspond to the the principal eigenvectors of $A^T A$ and AA^T, respectively. A key observation is that a single update to the user access traffic corresponds to a perturbation to the A matrix. Depending on the weight function selected, it can perturbate a single element or a row of A. In either case the perturbation is local. This perturbation will cause variation to the principal eigenvector of $A^T A$ (and AA^T). If we can find the relationship between the variation of x and y and the perturbation to A, we can check each update to see if it will cause too much variation to x and y. If the change is within acceptable precision, we can postpone applying the update thus avoiding running HITS for it. When the accumulated updates cause too much perturbation, we apply them together and run HITS once. This is essentially an approximation to HITS that amortizes its cost across multiple updates. We denote such an algorithm Online HITS. Another advantage of this approach is that service of user queries and updating A and running of HITS can be made separate. The system can update A and run HITS in background and continue servicing user queries with old results that we are confident to be within certain range from the the latest ones. Users can enjoy nonblocking service.

Similar issues have been discussed in the context of stability of the HITS algorithm [16, 17]. However, there is a subtle but significant difference between our approach and theirs: we are not concerned with the incompleteness of our data or the stability of the results. For us, the everlasting accumulation of data is an inherent feature of our system and the results we produce are the "best guess" based on the data we have so far. It is perfectly alright for the results to undergo dramatic change, which reflects the update of the system's knowledge about the world. Rather, we are interested in the bound of the change so that we can perform the tasks more efficiently. In addition, the conclusions in [16, 17] only apply to unweighted graphs represented by adjacency matrices. The theorem we describe below is applicable to any weighted graph.

Online HITS is based on the following theorem:

THEOREM 4.1. *Let $S = A^T A$ be a symmetric matrix. Let a^* be the principal eigenvector and δ the eigengap*[1] *of S. Let E_S be a symmetric perturbation to S. We*

[1]Eigengap is defined to be the difference between the largest and the second largest eigenvalues.

use $\|\cdot\|_F$ *to denote the Frobenius norm*[2]*. For any* $\epsilon > 0$*, if* $\|E_S\|_F \leq min\{\frac{\epsilon\delta}{4+\sqrt{2}\epsilon}, \frac{\delta}{2\sqrt{2}}\}$*, then the principal eigenvector* $\tilde{a}^*$ *of the perturbed matrix* $\tilde{S}$ *satisfies*

$$\|a^* - \tilde{a}^*\|_2 \leq \epsilon$$

This theorem gives us a way to test the perturbation and bound the error on the principal eigenvector. The proof is similar to that presented in [17] and is given in appendix.

There are two subtle issues that need to be addressed before we can use this theorem to construct an online HITS algorithm, namely the computations of eigengap δ and perturbation $\|E_S\|_F$. They have to be performed efficiently otherwise the cost of computing them would offset the saving of not running HITS. They will be addressed in the following subsections.

4.2 Computation of Eigengap A straightforward way of computing eigengap δ is to calculate λ_1 and λ_2, the largest and the second largest eigenvalues, and take the difference. The original HITS algorithm is essentially a power method to compute the principal eigenvector of S. It can be revised easily, without adding complexity, to produce λ_1 and λ_2 as byproducts. Two modifications to the original HITS algorithm are introduced:

1. Instead of finding only the principal eigenvector, find the two eigenvectors corresponding to λ_1 and λ_2. This can be done by using the "block power method" ([23], pp. 289). Concretely, start with two orthogonal vectors, multiply them all by S, then apply Gram-Schmidt to orthogonalize them. This is a single step. Iterate until they converge.

2. HITS normalizes the vector at each step to unit length. This is not necessarily the only choice to ensure convergence. Instead, we normalize each vector by dividing them by their first non-zero element. They still converge to the two eigenvectors and the scaling factors converge to λ_1 and λ_2 ([23], pp. 289).

The above modifications introduce extra computation of one eigenvector and Gram-Schmidt orthogonalization. The former doubles the work of HITS and the latter is $O(n)$. The total complexity is the same as HITS: $O(mn)$.

[2]The Frobenius norm of a matrix X is defined by $\|X\|_F = (\sum_i \sum_j (X_{ij})^2)^{1/2}$

4.3 Upper Bound of $\|E_S\|_F$ Let E be perturbation to matrix A (This is our update to the graph). Then $\tilde{A} = A + E$ and $\tilde{S} = (A+E)^T(A+E) = A^TA + A^TE + E^TA + E^TE$. Let $E_S = A^TE + E^TA + E^TE$. We know for Frobenius norm (actually for any norms) $\|X+Y\|_F \leq \|X\|_F + \|Y\|_F$. So $\|E_S\|_F \leq 2\|A^TE\|_F + \|E^TE\|_F$. This bound involves matrix multiplication which we try to avoid. Note that the purpose of our online HITS is to postpone running the algorithm so that we can save some computation. This means that we will accumulate a number of updates (since the last time we update A and run HITS). Even though each single update is local and involve only one element or one row of A, all the accumulated updates will affect a number of A's elements. This means E can be sparse but unlikely to have only single non-zero element or a row. Let $E(t)$ be the accumulated unapplied update matrix after we observed tth update (we reset the counting each time we apply updates). $E(t) = E(t-1) + \Delta(t)$ where $\Delta(t)$ has only one non-zero element or row. We have

$$\|E_S(t)\|_F \leq 2\|A^TE(t)\|_F + \|E(t)^TE(t)\|_F \tag{4.3}$$

where

$$\begin{aligned} \|A^TE(t)\|_F &= \|A^T(E(t-1) + \Delta(t))\|_F \\ &\leq \|A^TE(t-1)\|_F + \|A^T\Delta(t)\|_F \end{aligned} \tag{4.4}$$

and $\|E(t)^TE(t)\|_F =$

$$\begin{aligned} & \|(E(t-1) + \Delta(t))^T(E(t-1) + \Delta(t))\|_F \\ = \; & \|E(t-1)^TE(t-1) + E(t-1)^T\Delta(t) \\ & +\Delta(t)^TE(t-1) + \Delta(t)^T\Delta(t)\|_F \\ \leq \; & \|E(t-1)^TE(t-1)\|_F \\ & +2\|E(t-1)^T\Delta(t)\|_F + \|\Delta(t)^T\Delta(t)\|_F \end{aligned} \tag{4.5}$$

The three equations above give us a way to compute the upper bound on $\|E_S\|_F$ recursively. Namely we can keep running updates on the upper bounds of $\|A^TE(t-1)\|_F$ and $\|E(t-1)^TE(t-1)\|_F$ using Equation 4.4 and 4.5, respectively, and add to them the other terms in the equations to get new upper bounds of next step.

When a single update involves only one element of A, $\Delta(t)$ has a single non-zero element. Let $\Delta_{ij}(t)$ be the non-zero element of $\Delta(t)$, then

$$\begin{aligned} \|A^T\Delta(t)\|_F &= \Delta_{ij}(t)\|A_{i*}\|_2 \\ \text{and } \|E(t-1)^T\Delta(t)\|_F &= \Delta_{ij}(t)\|E(t-1)_{i*}\|_2 \end{aligned} \tag{4.6}$$

where A_{i*} and $E(t-1)_{i*}$ are the ith row of A and $E(t-1)$, respectively.

There are two ways to compute $\|A^T\Delta(t)\|_F$ or $\|E(t-1)^T\Delta(t)\|_F$: (1) keep the matrix $E(t-1)$ and use Equation 4.6; (2) use the maximum element of A or

$E(t-1)$ to estimate. (1) is accurate and involves $O(n)$ operations. (2) is fast (only scalar multiplication). The actual choice depends on application.

When an update changes a row of A, computing $\|A^T\Delta(t)\|_F$ and $\|E(t-1)^T\Delta(t)\|_F$ is more expensive and requires $O(n^2)$ operations and $\|\Delta(t)^T\Delta(t)\|_F = \|\Delta_{i*}(t)\|_2^2$ which is $O(n)$. This is at the same level of complexity as HITS but can be substantially cheaper to run because the latter takes a number of iterations to converge while the former needs to run only once. Kleinberg reported that the typical number of iterations for HITS to converge is 20 [12]. If the cost is still too high to accept, there are two ways to alleviate: (1) Frobenius norm has the property $\|AB\|_F \leq \|A\|_F\|B\|_F$. $\|A^T\Delta(t)\|_F$ and $\|E(t-1)^T\Delta(t)\|_F$ can be reduced to scaler multiplication (with loss of "tightness"); (2) the computation can be made to be distributed across all clients, as described in Section 6.

4.4 The Algorithm Putting all these together, we summarize the Online HITS algorithm in this section. In the following, we assume there is a procedure Gram-Schmidt that, given a matrix M, orthogonalizes its column vectors using Gram-Schmidt process ([23], pp. 129). We also assume there is a process that monitors the data and invokes our algorithm with perturbation when it sees an update.

Let $z_n = (1, 1, \ldots, 1)^T \in R^n$. Let $z_n^\perp \in R^n$ be the vector that is orthogonal to z_n and has the same length. $\Delta \in R^{n\times m}$ is the perturbation caused by a single update. ϵ is the required precision. Let $x[1]$ be the first non-zero element of vector x. We keep global variables $\|E_s\|_F, \|A^TE\|_F$ and $\|E^TE\|_F$. To make it concise, we use matrix computations in the pseudocode. However, it is clear that they can either be implemented together with HITS iterations, or only require operations on small number of the elements of the matrices involved, as described in Section 4.3.

The two main procedures are NewHITS and OnlineHITS. NewHITS is the modified version of HITS algorithm that performs block power iterations on two vectors and compute eigengap. Note that A^TA and AA^T share the none-zero eigenvalues so only one eigengap is needed. OnlineHITS is called on each update. It checks whether all the accumulated updates would cause large perturbation to the ranking. If so it will apply the updates and invoke NewHITS. Otherwise it returns the ranking from previous round. These two procedures are listed below.

```
NewHITS(A, ε)
    A ∈ R^{n×m}
    x ← z_m, x_⊥ ← z_m^⊥
    y ← z_n, y_⊥ ← z_n^⊥
    Do
        x ← Ay, x_⊥ ← Ay_⊥
        y ← A^T x, y_⊥ ← A^T x_⊥
        [x, x_⊥] ← Gram-Schmidt([x, x_⊥])
        [y, y_⊥] ← Gram-Schmidt([y, y_⊥])
        δ ← x[1] − x_⊥[1]
        x ← x/x[1], x_⊥ ← x_⊥/x_⊥[1]
        y ← y/y[1], y_⊥ ← y_⊥/y_⊥[1]
    Until error < ε
    Return (x, y, δ)

OnlineHITS(Δ, ε)
    ‖A^T E‖_F ← ‖A^T E‖_F + ‖A^T Δ‖_F
    ‖E^T E‖_F ← ‖E^T E‖_F + 2‖E^T Δ‖_F + ‖Δ^T Δ‖_F
    ‖E_s‖_F ← 2‖A^T E‖_F + ‖E^T E‖_F
    E ← E + Δ
    If ‖E_s‖_F > Tol
        A ← A + E
        [x, y, δ] = NewHITS(A, ε)
        E ← 0
        ‖A^T E‖_F ← 0
        ‖E^T E‖_F ← 0
        ‖E_s‖_F ← 0
        Tol = ϵδ / (4+√2ϵ)
    Endif
    Return (x, y)
```

5 Evaluation

Compared to HITS, OnlineHITS is at the same complexity level. However, its advantage lies in the hope that the updates may not cause too much perturbation to the ranking so that recomputation is avoided. In addition, the operations introduced for perturbation checking do not require iteration so they are substantially cheaper than HITS. The benefit gained by Online HITS depends on the stability of the system in face of perturbation, which is application-specific. We believe that in situations where data is accumulating, Online HITS is most likely advantageous. The intuition behind this belief is that the more data is accumulated, the less significant a new update would be to the overall ranking. Therefore there would be more opportunities to avoid update and running of HITS.

To evaluate how well Online HITS performs, we implemented the algorithm and ran it on the Enron Email Dataset [6]. We used some of the useful mappings created by Andres Corrada-Emmanuel [7]. In particular, for each email, we used the mappings to find its author and recipients. As pointed out in [7], The Enron corpus contains some inconsistencies. In our test, we ignored emails that were mapped to multiple authors. Multiple

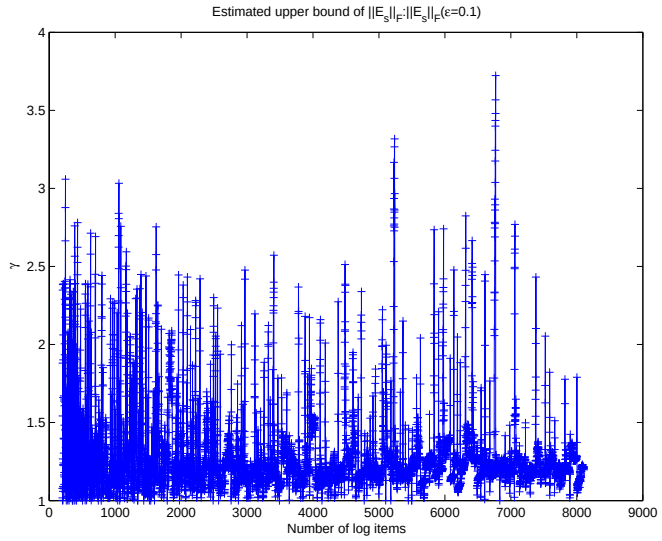


Figure 1: Approximation ratio.

recipients of a single email, however, are preserved.

This test can be thought of as "identifying the central figures" in the social network defined by the email communications. In constructing the graph, we simply used message count as the weight for each link between two users. Since one email may have multiple recipients, multiple links may be updated when an email is observed. There are a total of 150 users in the data set and our algorithm ranks them in "importance" according to their email communication.

An email is treated as a log item. Online HITS constantly monitors the log and performs operations as described in Section 4. A total of 8107 log items are observed. The precision is chosen to be $\epsilon = 0.1$ [3].

Note that we are not testing how well the ranking produced by Online HITS (or HITS) fits the "real" ranking which is a rather qualitative and subjective measure. Instead, we are examining Online HITS's algorithmic properties and how it performs more efficient than original HITS in a dynamic system.

The results of our test are shown in the following figures.

Figure 1 plots the ratio of the estimated upper bound of $\|E_s\|_F$ and its actual value. I.e. for each update $\gamma = (\|2A^T E\|_F + \|E^T E\|_F)/\|E_s\|_F$. It shows how tight the upper bound given in Section 4.3 is. The number varies as updates accumulate and are applied, but never exceeds 3.8.

Figure 2(a) shows, for each update, the actual perturbation $\|E_s\|_F$, the upper bound we estimated based on the method of Section 4.3, and the tolerance as specified by Theorem 4.1. Although the details are not easily discernable due to the large number of data points, it clearly shows the general trend of these measures, i.e. the tolerance grows as the data accumulates and allows for more and more perturbation while maintaining the given precision. Figure 2(b) enlarges one area of (a) to show the details. This area lies between data item 5965 to 6078. The horizontal line segments of red dots represent the intervals where the perturbation is within tolerable range and no update is applied. This particular line in Figure 2(b) demonstrates around 113 updates for which the NewHITS needs not to be invoked, i.e., a saving of 113 rounds of HITS computation. Similar savings can also be observed in other areas of Figure 2(a).

Figure 3 shows the rankings of "top" 10 users in the data set [4]. Both the actual ranking of each user (obtained by running HITS at each update) and the approximation produced by OnlineHITS are plotted. Note that in Figure 3(a) the rankings of the top 5 users are so close that their results appear in the figure almost as a single curve (the curve on the top). Preliminary investigation uncovered that they are all involved in a large number of error messages (one of them is the sender and the rest recipients) and, as the HITS algorithm discovered, they share similar roles in terms of their email communication pattern in the data set. Our algorithm discovers this structure as well. The estimated rankings are so close to the actual ones that it is difficult to distinguish them in Figure 3(a). Figure 3(b) enlarges part of (a) for clarity. It shows that the estimated rankings closely track the actual ones even when no recomputation is performed.

Our test demonstrates the substantial advantage of OnlineHITS when applied to an actual data corpus. We believe it is applicable to other dynamic environments as well. In particular, for systems that do not have a clearly marked leisure period (e.g. a system serving users from all time zones around the world), simply "running HITS at night" will not work. Our algorithm can provide an accurate estimate on the perturbation a update can cause and offers precise ranking in real-time.

6 Privacy Preserving Online HITS

The algorithm described in previous sections addresses the dynamic and real-time response issues of using access patterns in link analysis. However, in many situations, a naive implementation of the algorithm

[3] The choice of the precision depends on the application and the data. As we will observe later, the rankings of individual users in the Enron Email Dataset are quite "far" from each other and a larger ϵ can be used without affecting their relative standings. The result will be more saving in computation.

[4] For privacy reasons the names of the users are withdrawn.

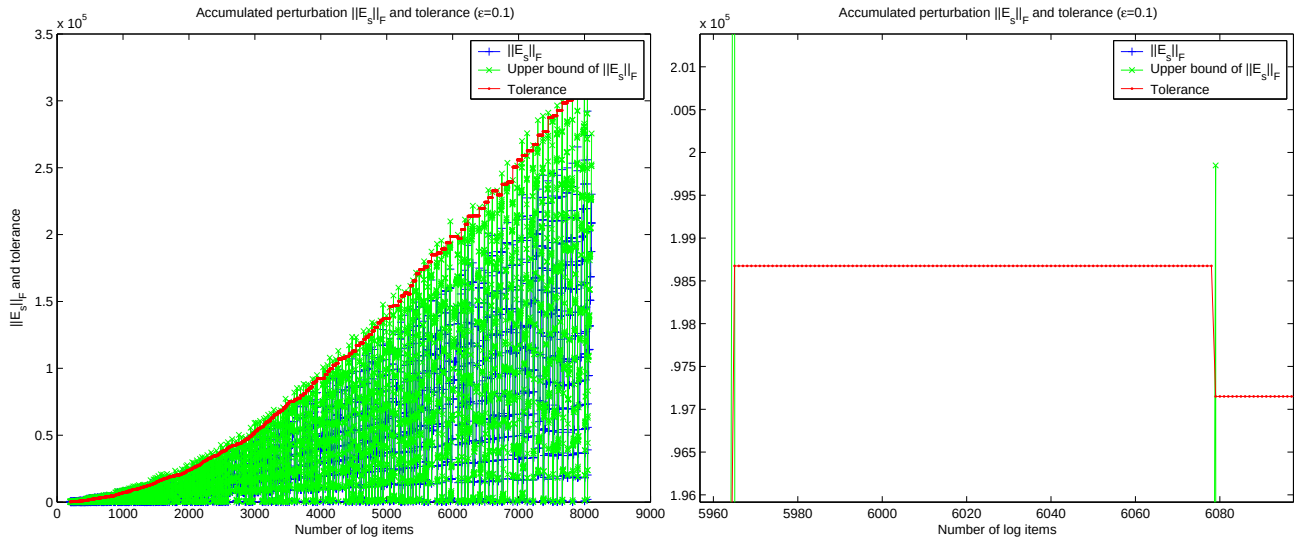


Figure 2: Accumulated perturbation $\|E_s\|_F$ and tolerance.

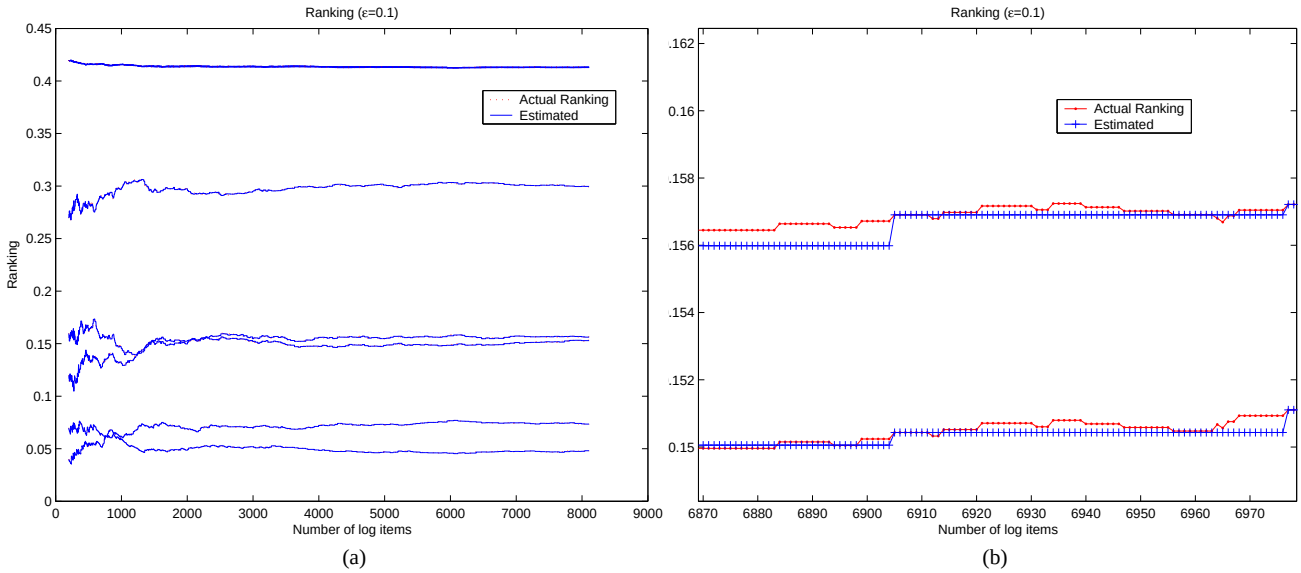


Figure 3: Ranking.

has severe privacy implications. In most applications, the weight on each edge of the graph represents the "rating" or "preference" of a user to the documents or other user and is gathered via client side logging. Such information is quite personal and exposing it would jeopardize the privacy of users thus hindering the acceptance of our system. If implemented directly, the online HITS algorithm would require the server running the algorithm be able to see all the data and involve substantial amount of network communication. In most situations trusting the server or network is not acceptable.

Fortunately this problem can be solved with cryptographic techniques. The general idea is that we can use encryption to protect user data and perform computations on encrypted data. Only aggregate data is made public and individual data are transmitted across the network in encrypted form. Such scheme has been proven to be sound and feasible, with satisfying performance. In [2], Canny proposed a privacy preserving collaborative filtering scheme that performs iterative SVD using the conjugate gradient method of Polak-Ribiere [20]. The basic building blocks, however, are homomorphism and threshold decryption which allow one to compute sums without disclosing summands. And this is exactly all that is needed to perform Online HITS. [2] gives complete description of the scheme and the proof of its soundness. Here we will sketch how it can be applied to make Online HITS distributed and privacy preserving.

In the following we will only consider the computation of document ranking, x. Expertise ranking is done in a similar fashion. We assume that most clients are honest and won't cheat or collude to pry about other's data. As in [2], we assume there is a write-once read-many (WORM) storage system where public data can be published. We also assume there is a tallier machine that performs addition on the data it receives. The tallier doesn't have to be a dedicated server. One of the clients can be designated as the tallier or its task can be made peer-to-peer.

6.1 Basics Several commonly used encryption schemes (RSA, Diffie-Hellman, ECC) have a useful property called homomorphism: if m_i is a message and $E(\cdot)$ is an encryption function, let g be a multiplicative group element, we can define a function $H(m) = E(g^m)$. This function satisfies

$$H(\sum_{i=1}^{n} m_i) = \prod_{i=1}^{n} H(m_i)$$

where multiplication is ring multiplication for RSA, or element-wise for DH or ECC. This allows one to obtain the encryption of a sum without revealing the summands.

Recovering the sum involves secret sharing and threshold decryption. The decryption key is not owned by any single party but secret-shared among all the clients. Pedersen's key generation protocol [18] or its variants/enhancements [9, 8, 3] can be used to securely generate the globally-known public key and distribute the secret shares of the private key among participants. At the end of their protocol, each client will have a share s_i of the decryption key, s, which could have been easily reconstructed from any set Λ of $t+1$ shares via Lagrange interpolation where t is a pre-defined threshold that is greater than the maximum number of dishonest clients in the system. This arrangement not only discourages clients from cheating but also introduces redundancy that makes the system robust – any $t+1$ shares of s can recover it. However, reconstructing s will effectively reveal the secret key to a single party thus rendering the scheme useless. Instead, we use threshold decryption on the ciphertext when decryption is desired. That is, each client decrypts the ciphertext with its share of the key and the result is a share of the decryption of the value. By putting these shares together, users can recover the encrypted value.

The value decrypted is not actually the sum of messages $\sum_{i=1}^{n} m_i$ that we are seeking, rather it is $g^{\sum_{i=1}^{n} m_i}$. Although recovering the sum requires taking discrete log, the value of sum will be small enough (10^6 to 10^9) so that a baby-step/giant-step approach is practical and the process can also be sped up by distributing it among many clients to be performed in parallel.

6.2 A Run of HITS The results of the tth iteration of HITS, x^t and y^t which are aggregate data, are made public. User i is responsible for his own rating of the documents (obtained via analyzing his document access pattern), namely the ith row of matrix A. Let $A_i^T = [a_{i1}, a_{i2}, \ldots, a_{im}]$ be that row. For the step $x^{t+1} = A^T y^t$, all that is involved from i is his own expertise ranking, y_i^t, and A_i^T. User i computes $y_i^t A_i^T$ and encrypts the vector and sends it to the tallier. The tallier, after receiving data from all users, produces the encryption of the sums of the ranking of each document by multiplying corresponding elements of the vectors. The resulting vectors are committed to the WORM. Users will read from WORM and perform threshold decryption to recover the values. This is x^{t+1}.

To compute y^{t+1} (which is Ax^{t+1}), user i computes $A_i^T x^{t+1}$ which is y_i^{t+1}, the ith element of vector y at iteration t, and publish it. If every user does this, the vector y^{t+1} can be obtained.

The iteration can stop when enough precision is achieved.

6.3 Perturbation Checking The scheme described in Section 6.2 shows a run of HITS, not Online HITS. To fit online HITS into such a scheme, we need to find a way to compute the perturbation, $\|E_S(t)\|_F$, with encrypted data or allow each user to compute with local data.

Recall that Equations 4.3, 4.4, 4.5 and 4.6 give us a way to update the upper bound of $\|E_S(t)\|_F$. The terms that need to be computed for each update are $\|A^T\Delta(t)\|_F$, $\|E(t-1)^T\Delta(t)\|_F$ and $\|\Delta(t)^T\Delta(t)\|_F$. Since for user i, $\Delta(t)$ has non-zero elements only in its ith row (and these numbers are obtained locally via his document access pattern analysis), $A^T\Delta(t)$ only involves the ith row of A, which the user maintains. Similarly, $E(t-1)^T\Delta(t)$ only involves the ith row of $E(t-1)$. In short, each user's update only involves his local data and it is straightforward to perform perturbation checking without disclosing private data: $\|E_S(t)\|_F$, $\|A^TE(t)\|_F$ and $\|E(t)^TE(t)\|_F$ are made public via the WORM storage and each user will update them using Equations 4.4, 4.5 and 4.6 with their local updates. When it is determined that it is time to update A, each user will update his own row and reset the perturbation records. All of them then collaboratively run the HITS algorithm as described in Section 6.2.

Note that we have actually killed two birds with one stone if we perform perturbation checking this way. Not only could we preserve user's privacy, we also distributed the computation among all users and parallelized the process.

There are some other issues in making such a scheme realistic such as dealing with dishonest users/tallier. Addressing them is out of the scope of this paper. [2] discusses such issues in detail and gives feasible solutions. In particular, it uses Zero Knowledge Proof (ZKP) to validate the data user and tallier produces so that they cannot excessively influence the results by cheating on their values.

7 Related Work

In [21], a set of heuristic graph algorithms are used to uncover shared-interest relationships among people, and to discover the individuals with particular interests and expertise, based on the logs of email communication between these individuals. The limitation with this approach is that experts are assumed to be communicating with fellow experts, which is not necessarily true in the real-world where experts may not be acquainted with one another, or may be rivals. Our approach does not assume any particular communication patterns between experts, and instead locate the experts based on their activities, e.g. if an expert accesses this set of authoritative documents, another person who accesses the same set is likely to be an expert as well.

The Referral Web [10, 11] is an interactive system for restructuring, visualizing and searching social networks on the World Wide Web. It constructs a graph of all users based on their email communication logs, which it uses to send a chain of referral requests until these requests reach an expert user. Like our Online HITS algorithm, Referral Web constructs the social network incrementally as it indexes the documents created and received by users. In contrast to our approach, however, the Referral Web raises possible privacy concerns because the chain of referrals inevitably reveal who someone down the chain knows to the user who initiates the search, unless individuals down the chain chooses not to forward the referral, in which case it becomes harder for the query to succeed.

Pirolli et al. [19] use a link-based approach like HITS to categorize webpages. It is similar to our weight-based algorithm in that users' access paths and metadata about webpages are used to construct the appropriate matrices. It differs significantly from ours in that while we use successive iterations to converge on our results, Pirolli et al. construct an activation network based on the strength of association between webpages and use the spread of activation in this network, starting from identified source webpages, to identify the webpages that exceed a threshold quantity of flow.

Carriere and Kazman's WebQuery system [4] rank webpages by considering the number of neighbors in the hyperlink structure that each webpage has. WebQuery performs link-based query post-processing to improve the quality of the results that it returns. In contrast, our incremental approach assumes that the hyperlink structure is highly dynamic, and postpones processing until the latest user-document accesses accumulate significant perturbation.

8 Conclusion

We extended the HITS hyperlink analysis algorithm to make it applicable to analyzing weighted graphs. Our generalizations are in two directions. First, we replaced the 0-1 valued hyper-link property to a non-negative valued weight function to model the users' behavior more accurately and proved its convergence. Second, we created an online eigenvector calculation method that can compute the results of mutual reinforcement voting efficiently in face of frequent updates by estimating the upper bound of perturbation and postponing applying the updates whenever possible. Both theoretical analysis and empirical experiments show that our generalized

online algorithm is more efficient than the original HITS under the context of dynamic data.

Finally we developed a secure variation of our online algorithm that solves the potential privacy issues that may occur when deploying large-scale access pattern-based document and authority ranking systems. Our scheme makes use of cryptographic techniques such as threshold decryption and homomorphic public-key encryption and distributes computation among users. Only aggregate or encrypted data are exposed. The scheme is also robust against a number of dishonest users up to a certain threshold.

Our extensions to Kleinberg's original HITS algorithm result in a generalized algorithm, Secure Online-HITS, that is practical for link analysis in scenarios such as collaborative work and online communities, in which there is no explicit link structure among nodes, and that users' access patterns of documents are highly dynamic, complex and should remain private.

9 Acknowledgements

The authors would like to thank the anonymous reviewers for their valuable comments.

References

[1] Sergey Brin and Lawrence Page, *The anatomy of a large-scale hypertextual web search engine*, in 7th World-Wide Web Conference, Brisbane, Australia, 1998.

[2] John Canny, *Collaborative filtering with privacy*, in IEEE Symposium on Security and Privacy, Oakland, CA, May 2002, pp. 45–57.

[3] John Canny and Stephen Sorkin, *Practical large-scale distributed key generation*, in Eurocrypt 2004, 2004.

[4] Jeromy Carriere and Rick Kazman, *Webquery: Searching and visualizing the web through connectivity*, in Proceedings of the International WWW Conference, 1997.

[5] S. Chakrabarti, B. Dom, D. Gibson, J. Kleinberg, P. Raghavan, and S. Rajagopalan, *Automatic resource list compilation by analyzing hyperlink structure and associated text*, in Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer, Addison-Wesley, 1989.

[6] William W. Cohen, *Enron email dataset*. http://www-2.cs.cmu.edu/~enron/.

[7] Andres Corrada-Emmanuel, *Enron email dataset research*. http://ciir.cs.umass.edu/~corrada/enron/.

[8] Pierre-Alain Fouque and Jacques Stern, *One round threshold discrete-log key generation without private channels*, Public Key Cryptography, (2001), pp. 300–316.

[9] Rosario Gennaro, Stanisław Jarecki, Hugo Krawczyk, and Tal Rabin, *Secure distributed key generation for discrete-log based cryptosystems*, Lecture Notes in Computer Science, 1592 (1999), pp. 295–310.

[10] H. Kautz, B. Selman, and A. Milewski, *Agent amplified communication*, in AAAI-96, Cambridge, Mass., 1996, MIT Press, pp. 3–9. Portland, Oreg.

[11] Henry Kautz, Bart Selman, and Mehul Shah, *Combining social networks and collaborative filtering*, Comm. ACM, 40 (1997), pp. 63–65.

[12] Jon M. Kleinberg, *Authoritative sources in a hyperlinked environment*, Journal of the ACM, 46 (1999), pp. 604–632.

[13] D. W. MacDonald and M. S. Ackerman, *Just talk to me: A field study of expertise location*, in ACM CSCW-98, 1998, pp. 315–324.

[14] A. Newell and P.S. Rosenbloom, *Matrix Computations*, Johns Hopkins University Press, 1989.

[15] A. Newell and P. S. Rosenbloom, *Mechanisms of skill acquisition and the law of practice*, in J.R. Anderson (Ed.), Cognitive Skills and their Acquisition (pp. 1-55). Hillsdale, NJ: Earlbaum, 1981.

[16] Andrew Y. Ng, Alice X. Zheng, and Michael Jordan, *Stable algorithms for link analysis*, in Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, ACM Press, 2001, pp. 258–266.

[17] Andrew Y. Ng, Alice X. Zheng, and Michael I. Jordan, *Link analysis, eigenvectors and stability*, in Proceedings of the 17th International Joint Conference on Artificial Intelligence, August 2001, pp. 903–910.

[18] T. Pedersen, *A threshold cryptosystem without a trusted party*, in Proceedings of EUROCRYPT '91, vol. 547 of Springer-Verlag LNCS, Springer, 1991, pp. 522–526.

[19] Peter Pirolli, James Pitkow, and Ramana Rao, *Silk from a sow's ear: Extracting usable structures from the web*, in Proc. ACM Conf. Human Factors in Computing Systems, CHI, ACM Press, 1996.

[20] E. Polak, *Computational Methods in Optimization*, Academic Press, 1971.

[21] M. F. Schwartz and D. C. M. Wood, *Discovering shared interests using graph analysis*, Comm. ACM, 36 (1993), pp. 78–89.

[22] G. W. Stewart and Ji-Guang Sun, *Matrix Perturbation Theory*, Academic Press, 1990.

[23] Gilbert Strang, *Linear Algebra and Its Applications, 2nd Edition*, Academic Press, 1980.

A Proof of Theorem 1

Proof. We use $\tilde{\cdot}$ to represent perturbated quantity. Suppose $S \in \mathbf{R^{n \times n}}$ is a symmetric matrix with principal eigenpair (λ^*, a^*), and eigengap $\delta > 0$. Let E_s be a symmetric perturbation to S such that $\tilde{S} = S + E_s$. By Theorem V.2.8 from matrix perturbation theory[22],

there is *some* eigenpair of $\tilde{S}(\tilde{\lambda}, \tilde{a})$ such that

$$\|a^* - \tilde{a}\|_F \leq \frac{4\|E_s\|_F}{\delta - \sqrt{2}\|E_s\|_F} \tag{1.7}$$

and

$$|\lambda^* - \tilde{\lambda}| \leq \sqrt{2}\|E_s\|_F \tag{1.8}$$

assuming the denominator in 1.7 is positive. Let $L \in \mathbf{R^{n-1 \times n-1}}$ be diagonal matrix containing all S's eigenvalues except λ^*. A bound similar to 1.8 holds:

$$\|L - \tilde{L}\|_F \leq \sqrt{2}\|E_s\|_F \tag{1.9}$$

Let $\tilde{\lambda_2}$ be the largest eigenvalue in $\tilde{L}$. By Corollary IV.3.6 of [22], Equation 1.9 implies

$$\tilde{\lambda_2} \leq \lambda_2 + \sqrt{2}\|E_s\|_F \tag{1.10}$$

Since $\|E_S\|_F \leq \frac{\delta}{2\sqrt{2}}$, Equations 1.8 and 1.10 ensures that $\tilde{\lambda} > \tilde{\lambda_2}$, i.e. $(\tilde{\lambda}, \tilde{a})$ is indeed the principal eigenpair of $\tilde{S}$. Also this will ensure the denominator in 1.7 is indeed positive.

Given any $\epsilon > 0$, if $\|E_S\|_F \leq \frac{\epsilon\delta}{4+\sqrt{2}\epsilon}$, then $\frac{4\|E_s\|_F}{\delta - \sqrt{2}\|E_s\|_F} \leq \epsilon$ thus we have $\|a^* - \tilde{a}\|_F \leq \epsilon$.

Mining Social Network from Spatio-Temporal Events

Hady W. Lauw* Ee-Peng Lim* Teck-Tim Tan† Hwee-Hwa Pang‡

Abstract

Knowing patterns of relationship in a social network is very useful for law enforcement agencies to investigate collaborations among criminals, for businesses to exploit relationships to sell products, or for individuals who wish to network with others. After all, it is not just what you know, but also whom you know, that matters. However, finding out who is related to whom on a large scale is a complex problem. Asking every single individual would be impractical, given the huge number of individuals and the changing dynamics of relationships. Recent advancement in technology has allowed more data about activities of individuals to be collected. Such data may be mined to reveal associations between these individuals. Specifically, we focus on data having space and time elements, such as logs of people's movement over various locations or of their Internet activities at various cyber locations. Reasoning that individuals who are frequently found together are likely to be associated with each other, we mine from the data instances where several actors co-occur in space and time, presumably due to an underlying interaction. We call these spatio-temporal co-occurrences events, which we use to establish relationships between pairs of individuals. In this paper, we propose a model for constructing a social network from events, and provide an algorithm that mines these events from the data. Experiments on a real-life data tracking people's accesses to cyber locations have also yielded encouraging results.

Keywords

social network, spatio-temporal data mining, link analysis

1 Introduction

Social network describes a group of social entities and the pattern of inter-relationships among them. What the relationship means varies, from those of social nature, such as kinship or friendship among people, to that of transactional nature, such as trading relationship between countries. Despite the variability in semantics, social networks share a common structure in which social entities, generically termed *actors*, are inter-linked through units of relationship between a pair of actors known as: *tie*, *link*, or *pair*. By representing actors as nodes and ties as edges, social network can be represented as a graph.

A constructed social network can be analyzed for many useful insights. For instance, the important actors in the network, those with the most connections, or the greatest influence [10, 17], can be found. Alternatively, it may be the connection paths between actors that are of interest. Analysts may look for the shortest paths [25], or the most novel types of connections [13]. Sometimes, the focus may even be on finding subgroups, subsets of the network that are especially cohesive or interesting [3, 15].

Knowledge of social networks is useful in various application areas. In law enforcement concerning organized crimes such as drugs and money laundering [25] or terrorism [12], knowing how the perpetrators are connected to one another would assist the effort to disrupt a criminal act or to identify additional suspects. In commerce, viral marketing exploits the relationship between existing and potential customers to increase sales of products and services [10, 17]. Members of a social network may also take advantage of their connections to get to know others, for instance through web sites facilitating networking or dating among their users [5].

Despite its many uses, social network is difficult to construct if only because a tie between a pair of actors is a property of the pair, rather than inherent to either actor. Collecting data on n actors quickly degenerates into finding the properties of $\frac{n(n-1)}{2}$ pairs of actors. Furthermore, the classical means of collecting such data by social scientists, though done carefully and reliably, are painstaking and time-consuming, involving questionnaires, interviews, direct observations, manual sifting through archival records, or various experiments [23]. This is fine for research studies experimenting on a small, controlled group of actors. However, wide application of social network analysis requires the ability to construct a large social network quickly, which can be achieved through computational methods capable of dealing with a huge amount of data.

In this paper, we look at computationally mining

*School of Computer Engineering, Nanyang Technological University, Singapore.

†Centre for IT Services, Nanyang Technological University, Singapore

‡Institute for Infocomm Research, Singapore

social network from spatio-temporal data. Each unit of such data has an associated location and time. Assuming that each data unit can also be attributed to a specific individual, the subset of data for an individual describes the series of locations visited by the individual over time. For example, such data may be obtained by tracking physical locations of moving objects, or by logging cyber locations visited by Internet users. Taking it a step further, we propose using spatio-temporal co-occurrence as a basis for inferring association between people. It is intuitive to think that co-occurring items may be related in some way, just as thunder's always following lightning tells us that they are somehow related. In this context, spatio-temporal co-occurrence is roughly defined as occurring together in space and time. By taking into account the frequency and the intensity of co-occurrences among people as they move around, we believe some knowledge about their relationships can be mined from the data.

Before stating the problem in earnest, we first enumerate our assumptions on the characteristics of data that we are dealing with. For a database $\mathcal{D}$, each tuple $d \in \mathcal{D}$ has the form of $d = \langle a, t, s \rangle$, where $d.a$ identifies an actor uniquely and $d.s$ indicates the location of this actor at time $d.t$. Though in reality seamlessly continuous, time is expressed as discrete values at a particular granularity (e.g., seconds). Furthermore, it is assumed that each data unit may be generated anytime, rather than only at strictly regular intervals as found in time series. Meanwhile, we model space as a collection of semantic locations, which may be physical locations such as rooms and buildings or cyber locations such as web addresses and domains. It is more practical to assume a semantic rather than a more refined coordinate space, which would have been more difficult to record accurately and would have required a mapping to correlate a coordinate to an actual location. Small-scale efforts to track locations, such as within building complexes, would likely settle for semantic location as it would be both easier and more useful to know that a person is at a particular room than at a given xyz coordinate. A semantic location may be expressed at several levels of granularity (e.g., room or building, web address or domain), and would also have a natural meaning indicating the purpose of the location, which would render a co-occurrence there even more meaningful.

We describe the problem in general terms as follows:

Given: *spatio-temporal database $\mathcal{D}$ as described*

Find: *social network graph $G(G_V, G_E)$, where:*

G_V is the set of nodes/actors in G

G_E is the set of edges/links in G based on spatio-temporal co-occurrences among actors

The problem as previously stated further spawns two subproblems:

1. *How are links between actors defined based on spatio-temporal co-occurrences?*
2. *How can such links be efficiently found?*

The rest of the paper is organized as follows. In Section 2, we survey various criteria used in mining social networks, and further explore the idea of co-occurrence. With respect to the first subproblem, in Section 3 we define a particular co-occurrence termed *spatio-temporal event* and describe how it could be used to infer links between actors. As a solution to the second subproblem, an algorithmic approach to mine social network based on events is presented in Section 4. Subsequently, Section 5 describes a real-life spatio-temporal data collected from web usage logs, and presents the experimental results on that data. Finally, in Section 6, we summarize our findings and suggest some directions for future work.

2 Related Work

Before we embark on a discussion on various ways of constructing social network, we first run through terms commonly used in social network literatures. As earlier mentioned, a node in a social network graph is termed an *actor*. A *tie* relates two actors. Like edges of a graph, ties could be directed or undirected, and they could be dichotomous (present or absent) or valued (weighted). There may be many types of ties (e.g., kinship, friendship) and the collection of all ties of the same type is a *relation*. *Social network* is a finite set of actors and all the relations among them. If all actors in a network are of the same type, the network is a *one-mode network*. Otherwise, a network with n types of actors is an *n-mode network*. These terms will be used throughout the rest of this paper.

2.1 Mining Social Network In addition to co-occurrence, these three criteria have also been used to infer ties between actors: self-report, communication, and similarity.

Self-report uses only links reported by individual actors. Such links are directed and naturally subjective. There could be cases where a claim of a tie is not reciprocated to the same extent, if at all. Classical tools like questionnaires and interviews are based on this principle [23]. Homepages or profile pages in community-centric sites such as LiveJournal weblogs [11] or Friendster networking site [5] commonly display a self-professed list of friends within the community. A similar idea is also present in the buddy list feature of Instant Messaging systems [18].

Communication, defined generally as transfer of information or resources, is common among socially related people. Inversely, evidence of communication may indicate association. Among others, such evidence may come from computer-mediated communication. Examples where the electronic trails of communication can be traced include emails [21], newsgroups [3], and Instant Messaging [18]. Links based on communication are directed, from the originator to the recipient.

Similarity has its foundation on the sociological idea that friends tend to be alike [6]. This leads to the premise that the more people have in common, the likelier it is that they are related. For example, homepages with similar textual content and linkages may represent a group of related individuals [1]. Other forms of similarity include having the same communication partners [21] and sharing the same opinions or areas of interest [17]. Similarity-based links are undirected.

Co-occurrence assumes that if several entities occur together more frequently than random chance alone would allow, they may be associated. Like similarity, it is by nature undirected and symmetric. The work on connection subgraph [8] uses a huge network whose ties identify pairs of people whose names are frequently mentioned together on the same webpages. Co-authorship networks, in turn, relate people who co-author the same publications together [10, 13].

Note that the above criteria for mining social network are seldom applied on spatio-temporal data. Of the four, co-occurrence is the most akin to such data as its meaning carries the sense of being together, possibly in space and time. This motivates us to pursue further the idea of spatio-temporal co-occurrence as a basis for inferring association.

2.2 Mining Co-occurrences With regards to time and space, there are four different ways to define co-occurrence: basic, when neither time nor space is considered; temporal and spatial, when only time or space is considered respectively; and spatio-temporal when both time and space are considered together.

Basic co-occurrence is mined from a database of discrete instances within which a few items co-occur. A major body of work on this type of co-occurrence is association rule mining [2]. For a given set of n items, $\mathcal{I} = \{i_1, i_2, \ldots, i_n\}$, a transaction is a discrete instance, uniquely identified by a *pid*, within which a subset of these items, $p \subseteq \mathcal{I}$, co-occur. The database to be mined is the set of all transactions $\mathcal{P}$. A pattern of co-occurrence involves a subset of items, $q \subseteq \mathcal{I}$, called an itemset, whose support is a function of $|\{p \in \mathcal{P} \mid q \subseteq p\}|$. If the support of q is beyond a given threshold, it is a frequent itemset, which is to say that members of q are deemed to be associated with one another.

Temporal co-occurrence does not assume that the data already has clearly-defined transactions. Instead, every tuple $\langle t, i\rangle$ is an item i occurring at time t, and subsets are derived using the time component of tuples. In the simplest case, two tuples $\langle t_1, i_1\rangle$ and $\langle t_2, i_2\rangle$ support an itemset $\{i_1, i_2\}$ if $|t_1 - t_2| \leq \delta$, for a given interval bound δ. Sequential patterns [4] and frequent episodes [16] not only care about the interval bound, but also the order at which items occur within the bound. Inter-transaction rules [14] would also demand the distance between occurrences of those items. Most strictly of all, time series patterns [7] specify an ordered series of items/values at regular intervals of time.

Spatial co-occurrence is aptly termed co-location. Each tuple $\langle s, i\rangle$ indicates that item i occurs at location s. Given the variety of spatial models [19], the notion of being co-located depends on the specific definition of space, from adjacent nodes in a graph space to items enclosed within a distance radius in a Euclidean space, but commonly captures the sense of being close by or neighboring. Another variation arises from how to define transaction-like instances over space. One way is to specify a reference feature (e.g., a lake), and treat each instance of that feature (and its neighboring items) as a transaction [9]. Alternatively, the space can be discretized by using a sliding window [20]. Yet another way is to materialize transactions wherever neighboring items are found, but constrain the multiplicity of the same item in many transactions [20].

Spatio-temporal co-occurrence deals with tuples with both space and time components. Despite the variability of spatial and temporal co-occurrences leading to the guess that there will be many ways to define spatio-temporal co-occurrence, current works in the area mainly focus on the time series approach. Spatio-temporal data is treated as a collection of time series of each item's wherebeing over time. Using time series similarity measures such as Euclidean [24] or LCSS [22] distance functions, the distance between two time series is evaluated. If it is below a certain threshold, the time series are considered similar enough, and the corresponding items are deemed to be co-occurring.

So far we have been mentioning co-occurrences of items, rather than actors. This is because the idea of spatio-temporal co-occurrence as indicative of association of social nature has not been much explored. Group pattern mining [24] is the closest to this direction, arguing that people who are consistently moving together may belong to a group. However, its focus is less on constructing a network formed by pairwise ties than on finding groups of increasing cardinality. Moreover, it assumes data in the form of time series of coordi-

nate locations, which leads to different formulations of the problem, and correspondingly to different solutions. In the next section, we propose a problem formulation based on irregular timing and semantic location that attempts to find pairwise ties between actors on the basis of spatio-temporal co-occurrence.

3 Mining Social Network from Events

3.1 Basic Events Just as an instance of co-occurring items is given the special term transaction in association rules, an instance of co-occurring actors is termed an *event* in social network terms. The work on inferring an association between actors through their participation in events is grounded in the affiliation network [23]. An affiliation network is a two-mode network, with a set of actors and a set of events connected by actor-event links. An event is any social collectivity of several actors, including conferences, games, social events, or meetings. An actor's affiliation to an event, by registration or attendance, establishes an actor-event link between the actor and the event.

By its act of bringing actors together, an event serves as conduit for resource transfer, or simply as a basis for interaction to take place. For example, conferences gather academicians around the world to exchange knowledge and build contacts. Linkages established through events can be interpreted in two ways. Firstly, an event enhances pairwise interactions between its members, in which case an event with n members gives rise to $\frac{n(n-1)}{2}$ actor-actor links. The second interpretation treats each event as a simultaneous linkage between all of its n members, much like a hyperedge connecting n vertices. Taking the first interpretation, which is more synchronous with most works in social network, an actor-actor tie between a pair of actors is said to exist if there is at least one event that the two actors are both affiliated to. Moreover, the number of such events can be taken as the weight of the tie. The collection of all such ties make up the social network.

In Figure 1, we give an example of an affiliation network, represented as a bipartite graph involving four actors $\{a_1, a_2, a_3, a_4\}$ and three events $\{e_1, e_2, e_3\}$. We have actors a_1 and a_2 affiliated to events e_1 and e_2, a_3 to e_2 and e_3, and a_4 to e_3. Based on their common affiliation to events, the actors can be linked in a social network as shown in Figure 2. Each actor-actor link indicates that two actors participate in at least one event together, and the weight of each link refers to the number of such events. Only a_1 and a_2 are linked by two events (e_1 and e_2), while the other pairs have only one event each. These figures illustrate how a two-mode affiliation network between actors and events can be transformed into a one-mode network of actors.

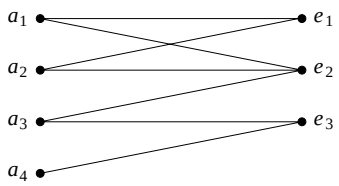


Figure 1: Two-Mode Affiliation Network

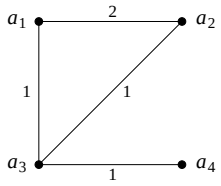


Figure 2: One-Mode Social Network

3.2 Spatio-Temporal Events Constructing such event-based networks as the above requires clearly-defined events gleaned from such sources as membership or attendance registers. Although spatio-temporal data as described does not carry information on events attended by actors, it can still tell us the spatio-temporal behavior of actors. We focus on one particular behavior: that actors may congregate together when engaged in social interactions. A corollary to that is that social events would produce spatio-temporal co-occurrences. Taking such co-occurrences as surrogates for events, we define a *spatio-temporal event* as a spatio-temporal co-occurrence that may have arisen from an underlying social interaction. Heretofore, we refer to nominal event as *basic event* and spatio-temporal event as just *event*.

Now we are ready to formally define an event. We adopt the notations for data as described before, where each tuple $d \in \mathcal{D}$ has the form of $\langle a, t, s \rangle$, identifying the location s of actor a at time t. Then, for a specified semantic location granularity and a time interval δ_{max}, an event is formally defined in the following way.

DEFINITION 3.1. *Event is a subset of tuples, $e \subseteq \mathcal{D}$, meeting the following conditions:*

- $\forall d_i, d_j \in e,\ d_i.s = d_j.s$,
  i.e., tuples are of the same location
- $\forall d_i, d_j \in e,\ |d_i.t - d_j.t| \leq \delta_{max}$,
  i.e., tuples are separated in time by at most δ_{max}
- $|\{d.a \mid d \in e\}| \geq 2$,
  i.e., tuples represent two or more actors
- *for any event $e' \subseteq \mathcal{D}$, $(e' \subseteq e) \vee (e \subseteq e') \Rightarrow (e = e')$,*
  i.e., each event is maximal

As required by the first two conditions, the semantic location granularity and time interval δ_{max} specify the constraints of a co-occurrence. Respectively, they indicate the furthest actors could be in space and time to still co-occur with each other. They could be as restrictive or as permissive as required to still render a co-occurrence meaningful in the sense of inducing some association among actors. Insisting on perfectly exact co-occurrences would be neither possible nor practical. Given the continuity of time and the limited sensitivity of even the finest measuring device, we cannot claim "at exactly the same time" with certainty anyway. While we use equality of locations to define co-occurrence, any location with non-zero area already implies a degree of tolerance. In any case, even if possible, exact co-occurrences might be rare. Furthermore, by allowing this tolerance to be varied, a suitable tolerance level can be chosen for the particular needs of the data.

The third condition requires that an event must concern more than one actor. It is obvious that for a spatio-temporal co-occurrence to be a surrogate for an actual interaction, it must involve at least two actors.

Finally, requiring each event to be maximal places a constraint on the multiplicity of tuples in being included in more than one event. Its purpose is to ensure, as much as possible, that each event stands for a single underlying interaction. Generally, events may overlap in terms of tuples, but they ought not to be subsets of another to avoid endless creation of events without gaining any additional information. The downside of this is that due to the constraint of δ_{max}, a long-running interaction may get split into several overlapping events.

Having defined event, we then enumerate some notations related to events. The set of all events defined over database $\mathcal{D}$ is denoted as $\mathcal{E}$. An event $e \in \mathcal{E}$ has several properties. The set of distinct actors represented by tuples in an event is its *actor set*, $e.\mathcal{A} = \{d.a \mid d \in e\}$, with size $|e.\mathcal{A}|$. An event's *start time*, $e.t^- = min_{d \in e}(d.t)$, and *end time*, $e.t^+ = max_{d \in e}(d.t)$, are the times of its earliest and latest tuples respectively. Correspondingly, its *duration*, $e.\delta = |e.t^- - e.t^+|$, is the distance between the two. The *area* $e.\triangle$ of an event measures the scope of its semantic location. We do not specify the exact form of this property, other than that for two locations, where one contains the other, the area value should be monotonic with respect to the granularity of the semantic location, i.e., the containing should have no smaller area than the contained. Lastly, its *weight* $e.w$ is a goodness measure related to the quality of relationship among actors of that event.

At this point we would like to note that perhaps with the exception of self-report, all other association criteria do not guarantee certainty in inferring association between actors. What they do is to mine evidence of association and assign a weight to each tie to indicate the likelihood of there being an actual association. Beyond a certain value where we feel confident enough about the existence of a tie, the weight may in turn assume the role of indicating the relationship strength of that tie. For affiliation network, every basic event is as good as any other as no effort is made to favor one over another. In our case, events possess some spatial and temporal information, which we will attempt to use to assign weights in ways that would boost the ability of events to both predict a relationship and measure the strength of that relationship. Towards this extent, we adopt the notions of *precision* and *uniqueness*.

Precision of an event refers to the quality of co-occurrence that defines the event with respect to tolerances in space and time. Intuitively, a co-occurrence at a finer granularity of space or time will also be valid at a coarser granularity. Besides being harder to achieve, the former is a more "exact", and thus a higher-quality, co-occurrence.

$$e.w_{p-s} = \frac{\frac{1}{e.\triangle}}{\max_{e' \in \mathcal{E}} (\frac{1}{e'.\triangle})} \tag{3.1}$$

Spatial precision of an event, denoted as $e.w_{p-s}$, measures how closely in space actors are from each other when participating in an event. This measure should be directly related to the granularity of the event's location, which in turn is related to the event's area $e.\triangle$. We define spatial precision as the inverse of an event's area, normalized with respect to the maximum such value among all events, as described by the equation Eq. 3.1. By this token, events held in smaller locations would be more precise than those in larger ones. The value of $e.w_{p-s}$ falls in the range of $(0, 1]$.

$$e.w_{p-t} = 1 - \frac{e.\delta}{(\delta_{max} + \delta_{unit})} \tag{3.2}$$

Temporal precision can similarly be based on duration $e.\delta$. Some may argue that very short durations are less important since they may have arisen from chance alone. That might have been valid if we know how long an actor stays at each location, which unfortunately we cannot know for certain given the assumption that the data is a set of snapshots, rather than a regular stream, of actors' locations. Instead, we take the reverse position that a shorter duration leads to a greater confidence that a co-occurrence has actually taken place. Besides, chance co-occurrences should be infrequent and can be removed accordingly. As such, temporal precision is defined as given in Eq. 3.2, giving a maximum value of 1 to events with perfect co-occurrence ($e.\delta = 0$). Addition of a unit of time δ_{unit} to the denominator is meant

to ensure a non-zero minimum value for cases where $e.\delta = \delta_{max}$. The value of δ_{unit} depends on the smallest division of time supported in the data, but in most cases we simply use $\delta_{unit} = 1$, assuming δ_{max} is expressed as a multiple of δ_{unit}. It follows that the range of $e.w_{p-t}$ falls in the range of $(0, 1]$.

Uniqueness is based on the idea that co-occurrence on a more unique premise is likely to indicate a stronger association. Unique items are deemed better because there is a lower probability of them being shared given their somewhat rarer occurrences. For instance, it has been suggested that commonly-shared features are weaker than unique features in predicting similarity-based association [1], or that novel, exclusive connections are more interesting than common ones [13].

$$e.w_{u-s} = 1 - \frac{|\{e' \in \mathcal{E},\ e' \neq e \mid e'.s = e.s\}|}{|\mathcal{E}|} \tag{3.3}$$

Spatial uniqueness refers to how unique the location of an event is among other events. Intuitively, if a location where not many other events take place is chosen, the interaction implied might also be of a different, and possibly more interesting nature. For an event e, its spatial uniqueness is given in Eq. 3.3. By counting only events other than itself, we ensure a non-zero minimum value such that $0 < e.w_{u-s} \leq 1$.

$$e.w_{u-t} = 1 - \frac{|\{e' \in \mathcal{E}, e' \neq e \mid e'.[t^-, t^+] \cap e.[t^-, t^+] \neq \emptyset\}|}{|\mathcal{E}|} \tag{3.4}$$

Temporal uniqueness, for all the same reasons, has an effect that is similar and parallel to spatial uniqueness. Instead of having a unique location, an event is temporally unique if it happens when relatively few other events are taking place. With a low level of background activity, it is an even lower probability that an event happens by coincidence. Furthermore, with such a judicious choice of time it is even likelier that the event is of a higher significance. However, in contrast to the semantic location case where overlap can be verified by equality, two events overlap temporally if they share at least a non-zero period of time. If the period of time covered by an event is denoted as an interval $e.[t^-, t^+]$, the function for temporal uniqueness is given in Eq. 3.4. As is the case with spatial uniqueness, we have $0 < e.w_{u-t} \leq 1$.

$$e.w = e.w_{p-s} \times e.w_{p-t} \times e.w_{u-s} \times e.w_{u-t} \tag{3.5}$$

Finally, we express an event's overall weight in Eq. 3.5 as the product of the above measures. Having non-zero value for each measure would prevent any one measure from nullifying the contribution of the other measures. Since each measure falls between 0 exclusive and 1 inclusive, the weight will also be in that range, $0 < e.w \leq 1$. Thus an event's weight can be interpreted as the probability that the event predicts an actual association between participating actors, or the strength of such a predicted association.

Dealing with semantic locations, we have defined spatial co-occurrence not in terms of distance interval, but in terms of a specified semantic location granularity. In reality, a database may have tuples with locations of varying granularity. For example, postal address has a home unit, city, state, and country. We may choose to restrict co-occurrence to the finest granularity only (e.g., home unit). However, what would be more practical is to allow co-occurrences to take place at various granularities, and to give events fair weights reflecting the weaker precision of a coarser granularity. Noting that locations of different granularities may subsume each other (e.g., home unit is contained in a city), we would not want to redundantly count events. In other words, two actors co-occurring in a city is redundant when we know they are in the same room. Towards this extent, we define a subevent-superevent relationship among events.

DEFINITION 3.2. *An event e_{sub} is a subevent of another event e_{sup}, or alternatively e_{sup} is a superevent of e_{sub}, if the following conditions are met:*

- $(e_{sup}.\triangle > e_{sub}.\triangle) \ \wedge \ (e_{sup}.s \text{ contains } e_{sub}.s)$
- $(e_{sup}.t^- \leq e_{sub}.t^-) \ \wedge \ (e_{sub}.t^+ \leq e_{sup}.t^+)$
- $e_{sub}.\mathcal{A} \subseteq e_{sup}.\mathcal{A}$

The first condition captures the sense that subevent-superevent relationship arises from differing location granularity. The latter two conditions are consequences of the first. By requiring co-occurrence at a finer granularity, a subevent is naturally more restrictive, and its duration and actor set are necessarily subsets of those of its superevent. Note that the relevance of these terms would come in later when we establish links based on events.

3.3 Event-based Links With some variation, we can derive a social network between pairs of actors based on spatio-temporal events in much the same way as that based on basic events. In affiliation network, a basic event is known for certain to be either present or absent. On the other hand, spatio-temporal events are inferred from the data and assigned a weight in the range of 0 to 1. If we take this weight as the probability that an event predicts an association, we may want to accept only events whose weight is above a certain threshold as capable of supporting links between actors.

DEFINITION 3.3. *An event e supports a link $\langle a_x, a_y \rangle$ between two actors, a_x and a_y, if $(\{a_x, a_y\} \subseteq e.\mathcal{A}) \wedge (e.w \geq min\_event\_weight)$, for a given threshold min_event_weight.*

For any given pair, there may well be more than one such event. We can then group together all such events as the *event set* of the pair. Furthermore, owing to the multi-granularity of semantic locations, we should take care to only include the most specific subevents supporting a linkage between the pair.

DEFINITION 3.4. *For a link $\langle a_x, a_y \rangle$, its event set is $\mathcal{E}_{\langle a_x, a_y \rangle} \subseteq \mathcal{E}$, such that:*

- $\mathcal{E}_{\langle a_x, a_y \rangle} = \{e \in \mathcal{E} \mid (\{a_x, a_y\} \subseteq e.\mathcal{A}) \wedge (e.w \geq min\_event\_weight)\}$
- $\forall e \in \mathcal{E}_{\langle a_x, a_y \rangle} \nexists e' \in \mathcal{E}_{\langle a_x, a_y \rangle}$, e' *is a subevent of* e

Greater cardinality of an event set means that more events support the association between the corresponding pair. Consequently, not only the link between the pair is more likely, it is also likely to be stronger. In order to factor this in the relationship strength of a pair, we define a link weight for a pair of actors $\langle a_x, a_y \rangle$ as the summation of the weight of the events in its event set, as given in Eq. 3.6. With that, we can then decide whether or not a link between a pair of actors exists.

$$\langle a_x, a_y \rangle.w = \sum_{e \in \mathcal{E}_{\langle a_x, a_y \rangle}} e.w \tag{3.6}$$

DEFINITION 3.5. *A link $\langle a_x, a_y \rangle$ between two actors, a_x and a_y, exists if $\langle a_x, a_y \rangle.w \geq min\_link\_weight$, for a given threshold min_link_weight.*

Keeping in mind that a social network is composed of links between pairs of actors, we restate the problem definition given previously as follows:

Given: *database $\mathcal{D}$, maximum duration δ_{max}, and thresholds min_event_weight, min_link_weight*

Find: *social network graph $G(G_V, G_E)$, where:*

$$G_E = \{\langle a_x, a_y \rangle \mid \langle a_x, a_y \rangle.w \geq min\_link\_weight\}$$
$$G_V = \{a \mid \exists \langle a_x, a_y \rangle \in G_E,\ a \in \{a_x, a_y\}\}$$

4 Algorithmic Approach

Since the database involved could be huge, in terms of the number of tuples and actors, the social network construction problem as posed in the above requires computational means to solve. Our proposed algorithm runs in two major phases. In the first phase, events are constructed from the database, and in the second phase, links are derived from those events.

ALGORITHM 4.1. Construction of Events

Input: database $\mathcal{D}$, time interval δ_{max}

Output: events $\mathcal{E}$

```
1: E = ∅,  E_cand = ∅,
2: for each tuple d ∈ D in the order of d.t do
3:    for each event e ∈ E_cand, (d.t − e.t⁻ > δ_max) do
4:       if (|e.A| > 1) ∧ (∄e' ∈ E, (e ⊆ e')) then
5:          E = E ∪ {e}
6:       end if
7:       E_cand = E_cand − {e}
8:    end for
9:    if ∄e ∈ E_cand, (e.s = d.s) ∧ (e.t⁻ = d.t) then
10:      create new event e = {d}
11:      E_cand = E_cand ∪ {e}
12:   end if
13:   for each event e ∈ E_cand, (e.s = d.s) do
14:      e = e ∪ {d}
15:   end for
16: end for
17: return E
```

The algorithm for the first phase is presented in Algorithm 4.1. The objective of this phase is to scan the database $\mathcal{D}$ and construct the set of events $\mathcal{E}$. Tuples of the database $\mathcal{D}$ are traversed in the chronological order. Recently created events that may still be affected by incoming tuples are first temporarily stored in $\mathcal{E}_{cand}$. This temporary store continually discards events whose temporal properties do not allow them to accept more tuples, i.e., when an event's duration would breach the limit of δ_{max}. Events with more than one actor and that are not just subsets of existing events in $\mathcal{E}$ are transferred into $\mathcal{E}$. A new event is created when a new location or a new timestamp is seen. Recent events in the temporary store $\mathcal{E}_{cand}$ of the same location as the incoming tuple are updated. Finally, the set of events $\mathcal{E}$ is returned as output of this phase. The time complexity of this phase is $O(|\mathcal{D}|)$, determined mainly by the outermost loop as the inner loops all concern $\mathcal{E}_{cand}$ whose cardinality is constrained by the value of δ_{max}.

Events created in the first phase are fed into the next phase, where the weights of these events are evaluated. The algorithm for the second phase is given in Algorithm 4.2. The first outermost loop iterates through the set of events $\mathcal{E}$. Each of the four measures, followed by the overall weight, of each event is computed. If an event's weight is beyond the threshold min_event_weight, it is eligible to support links among pairwise actors in its actor set. Each such pair is inserted into the set of candidate links $G_{E_{cand}}$. The algorithm also keeps the event set of each

pair updated and ensures that only the most specific subevents are used. At the end of the first outermost loop, we have the set of candidate links G_{Ecand} and the event sets of these candidate links. Subsequently, in the second outermost loop, the algorithm traverses through G_{Ecand}, first evaluating the weight of each candidate link and then verifying whether the weight is beyond the threshold *min_link_weight*. Such links are inserted into G_E, and the corresponding actors are inserted into G_V. As the computation of an event's weight may require traversal through $\mathcal{E}$, for instance to determine uniqueness the number of other events sharing similar spatial or temporal properties needs to be counted, the time complexity of the first outermost loop is $O(|\mathcal{E}|^2)$. The second outermost loop is clearly $O(|G_{Ecand}|)$. Hence this phase's time complexity is $O(|\mathcal{E}|^2 + |G_{Ecand}|)$.

ALGORITHM 4.2. Construction of Links

Input: events $\mathcal{E}$, *min_event_weight*, *min_link_weight*

Output: actors G_V, links G_E

1: $G_V = \emptyset,\ G_E = \emptyset,\ G_{Ecand} = \emptyset,$
2: **for** each event $e \in \mathcal{E}$ **do**
3: compute $e.w_{p-s}$, $e.w_{p-t}$, $e.w_{u-s}$, $e.w_{u-t}$
4: $e.w = e.w_{p-s} \times e.w_{p-t} \times e.w_{u-s} \times e.w_{u-t}$
5: **if** $e.w \geq min\_event\_weight$ **then**
6: **for** each pair $a_x, a_y \in e.\mathcal{A}$ **do**
7: $G_{Ecand} = G_{Ecand} \cup \{\langle a_x, a_y \rangle\}$
8: **if** $\nexists e' \in \mathcal{E}_{\langle a_x, a_y \rangle}, (e'$ subevent of $e)$ **then**
9: remove superevents of e from $\mathcal{E}_{\langle a_x, a_y \rangle}$
10: $\mathcal{E}_{\langle a_x, a_y \rangle} = \mathcal{E}_{\langle a_x, a_y \rangle} \cup \{e\}$
11: **end if**
12: **end for**
13: **end if**
14: **end for**
15: **for** each link $\langle a_x, a_y \rangle \in G_{Ecand}$ **do**
16: $\langle a_x, a_y \rangle.w = \sum_{e \in \mathcal{E}_{\langle a_x, a_y \rangle}} (e.w)$
17: **if** $\langle a_x, a_y \rangle.w \geq min\_link\_weight$ **then**
18: $G_E = G_E \cup \{\langle a_x, a_y \rangle\}$
19: $G_V = G_V \cup \{a_x, a_y\}$
20: **end if**
21: **end for**
22: return G_V, G_E

Combining the two phases is as easy as executing them in series. Initiated with database $\mathcal{D}$ as well as input parameters δ_{max}, *min_event_weight*, and *min_link_weight*, the combined algorithm outputs G_V and G_E, respectively the sets of nodes and edges of the desired social network graph G, at an overall time complexity of $O(|\mathcal{D}| + |\mathcal{E}|^2 + |G_{Ecand}|)$.

5 Experimental Results

5.1 Experimental Data For the experiments, we use a real-life data on webpages requested by wireless computer users at our university campus. The data is collected from firewall server logs over the whole month of August 2004. Each tuple contains a timestamp, a user login name, and a URL address. In total, there are about 4 million tuples, 2656 users, and 1.3 million URL addresses out of 58 thousand distinct URL domains. This data complies with the characteristics of spatio-temporal data that we expect. Actors are identified by their login names, which are anonymized to protect privacy. A tuple is generated whenever a URL request is made, and is timestamped up to the second ($\delta_{unit} = 1s$). URL addresses can be modeled as semantic locations and their directory structure corresponds to the multi-granularity of such locations. Here, we focus only on the URL domain level, and all the addresses are stripped down to their domains.

Though different from geographical locations, URL domains still have inherent semantic meaning in both the words that make up the domains as well as in the pages or sites that they represent. We figure that this semantic meaning would still render co-occurrences at such locations as potentially indicative of association between users. People do interact in the Internet and people visiting similar pages may have similar interests, may be collaborating on a task, may be influencing each other by recommending Internet resources, etc. All these carry a sense of association between people, the very thing we would like to mine.

5.2 Varying Parameters Through several experiments, we vary the input parameters to the algorithm to see the behavior or the properties of events and links that are generated. At any one time, we vary one parameter while fixing the rest. When fixed, the parameters would have the following values. We choose the maximum duration δ_{max} to be 2 hours which we deem a reasonable value for a meaningful co-occurrence at a URL domain. Expressing it in terms of δ_{unit}, we have $\delta_{max} = 7200s$. Next, we assume that all events should matter, so *min_event_weight* $= 0$. Meanwhile, we do not specify the value of *min_link_weight*, and first look only at candidate links, which are basically pairs of actors participating in at least one event together.

At first, we vary the size of the data along the chronological axis, while fixing the other parameters as mentioned in the above. Starting with a single day, we incrementally increase the input data, each time by adding a day's worth of data. Figure 3 shows the effect of increasing the number of tuples $|\mathcal{D}|$ to the number of events $|\mathcal{E}|$ and candidate links $|G_{Ecand}|$ generated.

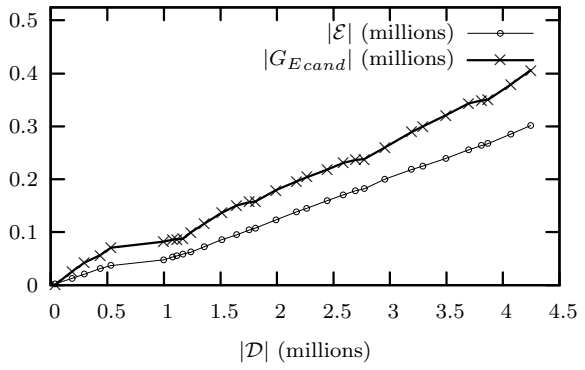


Figure 3: $|\mathcal{E}|$, $|G_{Ecand}|$ vs. $|\mathcal{D}|$

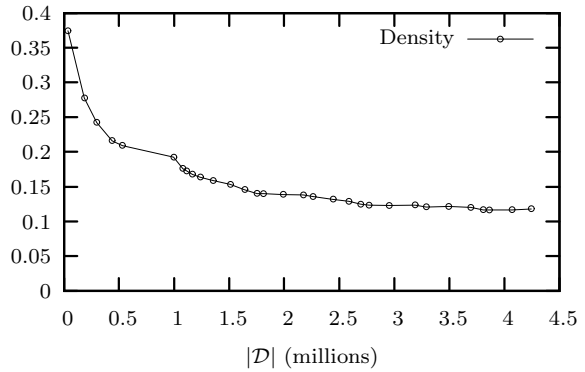


Figure 4: Density vs. $|\mathcal{D}|$

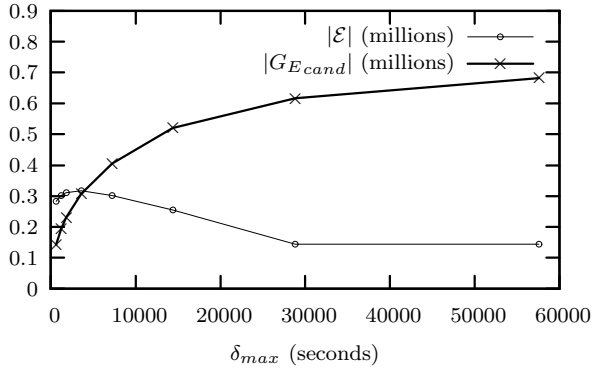


Figure 5: $|\mathcal{E}|$, $|G_{Ecand}|$ vs. δ_{max}

The latter two would have an effect on the algorithm's second phase's complexity. Clearly, there is an almost linear relationship between the data size varied along time and the number of events, which is actually quite intuitive as over an extended period of time, the rate at which events are taking place in real life should be relatively constant. Meanwhile, the steady increase in the number of candidate links is due to a different reason. Since not all actors are active all the time, extending the period of time being covered increases the likelihood that we catch their occurrences when they happen to be active. However, we expect that in the long run this number would level off as over a long enough period of time every actor would have had at least one event with every one of their acquaintances.

The increasing number of candidate links in Figure 3 is due to the increasing number of actors becoming active as data size increases. To take into account increases in the number of actors, we look at the density of the graph formed by the candidate links. Given n actors, the maximum possible number of links would be $\frac{n(n-1)}{2}$. Density of a graph is the fraction of the number of existent links over the maximum possible number [23]. For the graph formed from candidate links, the density value would in fact be $\frac{2\times|G_{Ecand}|}{n(n-1)}$. In Figure 4, we track this density as we increase the data size, which indirectly also increases the number of actors. We see that as the data size increases, the density at first decreases and then slowly converges to around 0.1, implying that for a large data size only about one-tenth of all possible links would be candidate links. This shows that the number of candidate links is related to the number of actors, and once the number of actors converges, so should the number of candidate links.

Next, we use the full data size, and again fix $min\_event\_weight = 0$ for the same reasons as the above. As δ_{max} is varied from 10 minutes to 16 hours, initially there is a growth in the number of events materialized, as shown in Figure 5. This is because a larger δ_{max} is more permissive that even tuples separated relatively widely in time can still form an event. Beyond a certain value of δ_{max}, the number of events begin to decline before leveling off as a very large δ_{max} results in several events of the same location being combined with one another to form a long-running event. In contrast, the number of candidate links continues to increase, though at a decreasing rate and eventually leveling off. Larger values of δ_{max} tend to be less restrictive in creating events, leading to more pairs having at least one common event.

Previously, no min_link_weight has been specified and we have only looked at candidate links. If specified, candidate links whose weight exceeds this

min_link_weight	$\|G_E\|$
0	406078
1	71866
5	5299
10	1569
20	421
30	176
40	85
50	44
60	25
70	13
80	7
90	3
100	2

Figure 6: *min_link_weight* vs. $|G_E|$

min_link_weight value would be included as links in the social network. Using the full data size, and parameter values $\delta_{max} = 7200s$ and *min_event_weight* $= 0$, we vary *min_link_weight* from 0 to 100 to get the number of links produced at each threshold value. Although we expect that the number of links ($|G_E|$) will be lower at higher threshold values, Figure 6 further shows that the drop in the number of links caused by increasingly higher thresholds is extremely precipitous. With 2656 actors, there could be up to $(\frac{1}{2})(2656)(2655)$ or 3.5 million links. Less than 12% of that number is supported by any event at all (*min_link_weight* $= 0$). By *min_link_weight* $= 20$, the number of links has dropped to hundreds. We recall that in affiliation network, a link is weighted by the number of basic events, and is deemed to exist if there is at least one basic event supporting that link. In our case, a link's weight is the sum of its supporting events' weights, with each event having a weight between 0 and 1. Setting *min_link_weight* $= 1$ would be equivalent to requiring at least one basic event to establish a link. In turn each *min_link_weight* value can be interpreted as the number of full events required to instill enough confidence that a pair of actors are actually related. There is a direct tradeoff between the confidence in links and the number of links that can be included in the social network graph.

Notably, with rare occurrences of links with very high weight while the vast majority of links have very low weight (0 to 1), the distribution of link weights seems to approximate the Zipfian [26] distribution, a distribution that has been shown by many other social networks as well [17].

Common Features	*Random-Pairs*	*Event-Pairs*
at least 0	100%	100%
at least 1	49%	90%
at least 2	12%	23%
at least 3	1%	3%

Figure 7: Demographic Similarity

5.3 Demographic Similarity Ideally, the links generated by the proposed event-based method can be verified to a high degree of confidence by gathering feedback from the concerned actors directly. Unfortunately, that has not been feasible in our case as there are strict restrictions on approaching the actors included in the data directly to protect their privacy. However, we have a limited demographic information about the actors. Relying on the idea that related actors tend to be similar (Section 2.1), we wish to check whether the event-based links that we have generated would show greater demographic similarity than links drawn at random.

The demographic features that can be obtained for each actor include her *major* (e.g., business, computer science), *status* (e.g., undergraduate, postgraduate, staff), and *year of entry* into the university. For each link between a pair of actors, we count the number of feature values the two actors have in common (from 0 to 3). For comparison, we draw two sets of links. *Random-Pairs* consists of 100 links formed by drawing a pair of actors at random from the pool of actors. *Event-Pairs* consists of 100 links with the highest link weights among the links generated by the proposed method run on the full data with parameters $\delta_{max} = 7200s$ and *min_event_weight* $= 0$. For each set, we count the number of links having at least 0 to 3 feature values in common. The results shown in Figure 7 confirm that at high threshold values, there tends to be a greater amount of demographic similarity in the event-based links than in the random links. While not spectacular by itself, *Event-Pairs* shows a not insignificant increase over *Random-Pairs*. On average, *Event-Pairs*' similarity percentages are about twice those of *Random-Pairs*.

To illustrate highly similar event-based pairs, in Figure 8 we use as examples the three pairs from the *Event-Pairs* set that have all three demographic features in common. Demography refers to the status, major, and year of entry of both actors in a pair. The first pair of actors, referred to as $\langle a_1, a_2 \rangle$, are both MBA students beginning in 2004. Events involving these actors include, but not exclusively, the given URL domains. The first two domains, those of Yahoo! India and Rediff.com (an India-based portal), indicate their Indian origin. The next two domains tell us their

Pairs	*Demography*	*Sample URL Domains*
$\langle a_1, a_2 \rangle$	Postgraduate (MBA) Business 2004	login.india.yahoo.com www.rediff.com www.carinfousa.com cdn.movies-etc.com
$\langle a_3, a_4 \rangle$	Postgraduate (Research) Biology 2003	www.ecallchina.com www.sohu.net nar.oupjournals.org www.ncbi.nlm.nih.gov
$\langle a_5, a_6 \rangle$	Postgraduate (Research) Civil Engin. 2003	eae.seu.edu.cn www.sciencedirect.com www.sina.com.cn xintv.xinhuanet.com

Figure 8: Highly Similar Event-based Pairs

common interests in car prices in the USA and in online movies. The second pair of actors, $\langle a_3, a_4 \rangle$, are both research students in biology beginning in 2003. The first two sample domains are China-based portals, again revealing their country of origin. Those are followed by domains belonging to the Nucleic Acid Research Journal and National Center for Biotechnology Information respectively, which suggest their similar research areas. The last pair of actors, $\langle a_5, a_6 \rangle$, are research students in civil engineering beginning in 2003. Both actors might have affiliation to South East University in China, as indicated by the first domain listed. Both have also used ScienceDirect, an online library portal, presumably for their research. Finally, the next two domains are again those of popular China-based portals. In these cases, we are fairly confident that actors in each pair are likely to know each other given such similar areas of interests, countries of origin, and demographic features. Furthermore, they also show that the event-based approach seems to be able to generate results that correlate with those from the similarity-based approach.

Rather than claiming the results above as absolute, we caution that the demographic information used to derive similarities is rather limited and that similarity on its own is not an authoritative method for verification. Nevertheless, we are still encouraged that the correlation between our proposed co-occurrence-based method with another, similarity-based method seems to indicate that our approach has a promising research potential.

6 Conclusion

In this paper, we introduce the problem of mining social network from spatio-temporal data. We propose using spatio-temporal co-occurrence as a basis for inferring associations of social nature. This is facilitated by our novel definition of spatio-temporal events, which we then use to derive event-based links between pairs of actors. After providing an algorithm that mines the desired event-based social network in two phases, we present our experiments on a real-life data on web usage logs collected at our own university. Comparison of the links produced by our proposed method and another, similarity-based method shows an encouraging result, especially keeping in mind that it has a real potential of generating large social networks from spatio-temporal data quickly for industrial or commercial uses.

There are many avenues for future works. Our current approach could be fine-tuned by investigating other factors that may help boost the quality of events and by learning from the results on different datasets. Faster algorithms that can deal with much larger data size or data streams would increase the utility of the proposed approach. The constructed social network can also be analyzed for useful patterns or insights such as temporal evolution or periodicity of relationships. Finally, we would also look at how patterns of mobility in spatio-temporal data, concerning speeds and sequence of locations traversed, may be used in mining social networks.

Acknowledgments

We would like to thank the Centre for IT Services, Nanyang Technological University, for providing us with the experimental data used in this paper, as well as the Agency for Science, Technology and Research (A*STAR) for partially funding the work presented in this paper through an A*STAR Graduate Scholarship.

References

[1] L. A. Adamic and E. Adar, *Friends and neighbors on the web*, Social Networks, 25 (2003), pp. 211–230.

[2] R. Agrawal and R. Srikant, *Fast algorithm for mining association rules*, VLDB, (1994), pp. 487–499.

[3] R. Agrawal, S. Rajagopalan, R. Srikant, and Y. Xu, *Mining newsgroups using networks arising from social behavior*, WWW, (2003), pp. 688–703.

[4] R. Agrawal and R. Srikant, *Mining sequential patterns*, ICDE, (1995), pp. 3–14.

[5] D. M. Boyd, *Friendster and publicly articulated social networking*, in Conf. on Human Factors and Computing Systems, (2004), pp. 1279–1282.

[6] K. Carley, *A theory of group stability*, American Sociological Review, 56 (1991), pp. 331-354.

[7] G. Das, K. Lin, H. Mannila, G. Renganathan, and P. Smyth, *Rule discovery from time series*, KDD, (1998), pp. 27–31.

[8] C. Faloutsos, K. S. McCurley, and A. Tomkins, *Connection subgraphs in social networks*, in Workshop on

Link Analysis, Counterterrorism, and Privacy (in conj. with SDM), (2004).

[9] K. Koperski and J. Han, *Discovery of spatial association rules in geographic information databases*, SSD, (1995), pp. 47—66.

[10] D. Kempe, J. Kleinberg, and E. Tardos, *Maximizing the spread of influence through a social network*, KDD, (2003), pp. 137–146.

[11] R. Kumar, J. Novak, P. Raghavan, and A. Tomkins, *Structure and evolution of blogspace*, CACM, 47 (2004), pp. 35–39.

[12] V. E. Krebs, *Mapping networks of terrorist cells*, Connections, 24 (2002), pp. 43–52.

[13] S. Lin and H. Chalupsky, *Unsupervised link discovery in multi-relational data via rarity analysis*, ICDM, (2003), pp. 171–178.

[14] H. Lu, L. Feng, and J. Han, *Beyond intratransaction association analysis: mining multidimensional intertransaction association rules*, ACM TOIS, 18 (2000), pp. 423–454.

[15] M. Mukherjee and L. B. Holder, *Graph-based data mining on social networks*, in Workshop on Link Analysis and Group Detection (in conj. with KDD), (2004).

[16] H. Mannila, H. Toivonen, and A. I. Verkamo, *Discovering frequent episodes in sequences*, KDD, (1995), pp. 210–215.

[17] M. Richardson and P. Domingo, *Mining knowledge-sharing sites for viral marketing*, KDD, (2002), pp. 61–70.

[18] J. Resig, S. Dawara, C. M. Homan, and A. Teredesai, *Extracting social networks from instant messaging populations*, in Workshop on Link Analysis and Group Detection (in conj. with KDD), (2004).

[19] S. Shekhar, S. Chawla, S. Ravada, A. Fetterer, X. Liu, and C. Lu, *Spatial databases - accomplishments and research needs*, IEEE TKDE, 11 (1999), pp. 45–55.

[20] S. Shekhar and Y. Huang, *Discovering spatial co-location patterns: a summary of results*, SSTD, (2001), pp. 236–256.

[21] M. F. Schwartz and D. C. M. Wood, *Discovering shared interests using graph analysis*, CACM, 36 (1993), pp. 78–89.

[22] M. Vlachos, G. Kollios, and D. Gunopulos, *Discovering similar multidimensional trajectories*, ICDE, (2002), pp. 673-684.

[23] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*, Cambridge University Press, 1994.

[24] Y. Wang, E. Lim, and S. Hwang, *On mining group patterns of mobile users*, DEXA, (2003), pp. 287–296.

[25] J. Xu and H. Chen, *Fighting organized crimes: using shortest-path algorithms to identify associations in criminal networks*, Decision Support Systems, 38 (2004), pp. 473–487.

[26] G. K. Zipf, *Human Behavior and the Principle of Least Effort*, Addison-Wesley, Boston, MA, 1949.

Unsupervised Name Disambiguation via Social Network Similarity*

Bradley Malin†

Abstract

Though names reference actual entities it is nontrivial to resolve which entity a particular name observation represents. Even when names are devoid of typographical error, the resolution process is confounded by both ambiguity, where the same name correctly references multiple entities, and by variation, when an entity is correctly referenced by multiple names. Thus, before link analysis for surveillance or intelligence-gathering purposes can proceed, it is necessary to ensure vertices and edges of the network are correct. In this paper, we concentrate on ambiguity and investigate unsupervised methods which simultaneously learn 1) the number of entities represented by a particular name and 2) which observations correspond to the same entity. The disambiguation methods leverage the fact that an entity's name can be listed in multiple sources, each with a number of related entity's names, which permits the construction of name-based relational networks. The methods studied in this paper differ based on the type of network similarity exploited for disambiguation. The first method relies upon exact name similarity and employs hierarchical clustering of sources, where each source is considered a local network. In contrast, the second method employs a less strict similarity requirement by using random walks between ambiguous observations on a global social network constructed from all sources, or a community similarity. While both methods provide better than simple baseline results on a subset of the Internet Movie Database, findings suggest methods which measure similarity based on community, rather than exact, similarity provide more robust disambiguation capability.

Keywords: Disambiguation, Social Networks, Random Walks, Multi-class Clustering

1 Introduction

Technological advances have sustained a continuing increase in our abilities to gather, store, and model information at the entity-specific level. With respect to entity-specific, or social, networks, the types of relationships which are learnable are vast and can provide detailed knowledge ranging from individual preferences to organizational structures. Yet, before knowledge regarding an entity or relationships between entities can be extracted from relational systems we must first attend to a more fundamental feature of data: correctness. Specifically, we must be able to decide when two pieces of data correspond to the same entity or not. Failure to ensure correctness can result in the inability to make inferences or the learning of false knowledge. The ability to decide when two or more pieces of data refer to the same entity is crucial not only for correct network construction and analysis, but to a wide range of critical processes, including data fusion, cleaning, profiling, speech recognition, and machine translation.

For surveillance and counterterrorism analysis, the relational data of interest is often made up of names, such that a vertex refers to a particular name and an edge specifies the relationship between two names. However, even when names are devoid of typographical errors, there are additional confounders to data correctness. First, there can exist name variation, where multiple names correctly reference the same entity. Second, there can exist name ambiguity, such that the same name correctly references multiple entities. While both problems must be accounted for, this paper concentrates on the basic aspects, and how to resolve, ambiguity. The basic question we ask is, how do you resolve which particular entity is referred to, or disambiguate, various observations of the same name?

Disambiguation is by no means a trivial feat, and the manner by which an individual makes the decision is often contingent on the available contextual clues as well as prior, or background, information. For example, when a reader encounters the name "George Bush", the reader must decide if the name represents "*George H.W. Bush*" - the 41st President of the United States of America, or "*George W. Bush*" - the 43rd president, or some other individual of lesser notoriety. How does one determine whom the name corresponds to? When the name is situated in a traditional communique, such as a news story, we tend to rely on linguistic and biographical cues. If the name is situated in the following sentence, "*George Bush was President of the*

*Partially supported by the Data Privacy Laboratory at Carnegie Mellon University and NSF IGERT 9972762 in CASOS.

†Institute for Software Research International, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, USA 15213-3890, malin@cs.cmu.edu

United States of America in 1989.", then, with basic knowledge of American history, it is clear the story refers to the elder "*George H.W. Bush*".

Though spoken conversations and written communications between entities are structured by known grammars there is no requirement for text-based documents to provide traditional semantic cues. One such counter scenario, which explicitly concerns social networks, occurs when documents are merely rosters that consist of nothing but names. [30] To relate information corresponding to the same entity in this type of environment, disambiguation methods must be able to leverage list-only information. Models employed in natural language processing [32], such as those available in the sentence regarding the American President, are not designed to account for this new breed of semantics.

There has been some headway made in the design of less structure dependent disambiguation methods. [6, 7, 21] However, these methods are often tailored to assumptions and characteristics of the environments where the references reside. For example, some methods leverage the covariates of references (i.e. the observation of two references in the same source) or require that social groups function as cliques. [6, 7] This model expects environments in which strong correlations exist between pairs or sets of entities, such that they often co-occur in information sources. While closely knit groups of entities provide an ideal scenario, it is not clear if such social settings manifest in the real world. In contrast, it is feasible, and intuitive, to leverage less directly observed relationships. This is precisely the route explored in this paper. We consider networks of the references in question, such that one can leverage "community" structures among entities. By studying communities of entities, we exploit relationships between entities which have minimal, or no, observed interactions. This is extremely powerful, since it allows for disambiguation when covariates are weak or the social network of entities is less centralized.

In this research paper, we investigate the degree to which disambiguation methods can be automated using relational information only. More specifically, given only a set of observations of names from information sources, such as webpages, can we construct an automated system to determine how many entities correspond to each particular name? Furthermore, can we determine which particular name observation corresponds to which underlying entity? The methods discussed in this paper are evaluated on a real world dataset derived from the Internet Movie Database (IMDB). Experimental findings from this research suggest that community similarity, which leverage indirect relationships, is more reliable for disambiguation than similarity methods which rely on direct relationships. In addition, we demonstrate that simple methods, such as those based on random walks can be applied towards estimating community similarity.

The remainder of this paper is organized as follows. In the following section, related research in linkage and disambiguation, including recent developments within the data mining community, is reviewed. In Section 3, the disambiguation methods which are applied in this research are formally introduced and defined. In Section 4, the IMDB dataset is summarized and the results of disambiguation experiments with this dataset are presented. Then, in Section 5, we consider some of the limitations of this research, discuss some of potential extensions, and consider some applications of social network-based disambiguation. Finally, in Section 6, the contributions of this research are summarized.

2 Background and Related Research

There exist a number of approaches that have been applied to disambiguation. In this section, we briefly review previous disambiguation research and where the work presented in this paper differs.

In general, disambiguation methods can be taxonomized on two features: 1) information type and 2) supervision. Information type specifies to whom data corresponds and there are two main types often used for disambiguation: a) personal and b) relational. Personal information corresponds to static biographical (e.g. *George H.W. Bush* was the 41^{st} President) and grammatical (e.g. *fall* used as a noun vs. as a verb) information. To leverage this information, disambiguation methods usually use sets of rules for discerning one meaning from another. In contrast, relational information specifies the interactions of multiple values or terms (e.g. *George H.W. Bush* collocates with *Ronald Reagan* whereas *George W. Bush* collocates with *Dick Cheney*).

The second taxonomizing feature is the supervision of the disambiguation process. In a supervised learning systems, each of the a disambiguation method is trained on labeled sample data (e.g. first sample corresponds to first meaning, second sample corresponds to second meaning, etc.). In an unsupervised learning system, methods are not trained, but instead attempt to disambiguate based on observed patterns in the data.

2.1 Personal Disambiguation. Word sense disambiguation methods initially gained momentum in natural language processing. Early computational methods tagged sentences with parts of speech and disambiguated words/phrases based on part of speech. [8, 19] With the incorporation of a database-backed model, IBM's "Nominator" system [33], uses phrase context

(e.g. punctuation, geographic position in sentence, and capitalization) in parallel with prior knowledge (e.g. known type of entity for names) for disambiguation. Names encountered by the system are matched to names whose context and knowledge have been previously specified. An alternative supervised method is to perform disambiguation using parallel corpora, such as in the cross-lingual context. [28]

Bagga and Baldwin [3] introduced an unsupervised disambiguation model based on sentence comparison for when prior knowledge is unknown. Sentence are parsed into vector-space summaries of words or concepts. Summary pairs are compared and similarity scores above a certain threshold are predicted as the same entity. Mann and Yarowsky [25] extend summaries to parse and structure biographical data, such as birth day, birth year, occupation, and place of birth. Once each name is associated with a simply biography, the name observations are clustered based on similarity of their biographies.

The recently developed "Author-ity" system, is an unsupervised system developed for database queries. Input is provided to this system as an author's name, in the form of last name and first initial. The system returns a list of scientific articles, authored by the name of interest, ranked in decreasing certainty of whether or not an article was authored by the same person. [31] Articles are ranked by performing a pairwise similarity of title, journal name, coauthor names, medical subject headings, language, affiliation, and prevalence of name in the database.

A drawback of personal information dependent methods is their lack of accountability for unstructured information. These methods require rules, grammars, and or multiple attributes for comparison.

2.2 Relational Disambiguation. An alternative approach for natural language disambiguation is based on a probabilistic model of word usage. Lesk [24] extended rule based models to account for the relationship of an ambiguous word with its surrounding words. He demonstrated that for an ambiguous word, overlap in the dictionary definitions' of surrounding text words can be used to disambiguate. Gale et. al. [14] demonstrated that the dictionary definitions are unnecessary provided a representative sample of word covariation was available. In their research, a Naïve Bayes classifier was trained for each ambiguous word and its surrounding words. Given a new word observation for disambiguation, the word was labeled with the definition of the max score classifier. Additional statistical models for using word and concept covariates have been studied. [9, 15, 16, 27, 34] A classifier based on covariance (i.e. the probability that a word occurs with another word) is trained for each meaning of the ambiguous word. For each new ambiguous word occurrence, a sense prediction is made based on which classifier the word, and its surrounding words, are most similar to.

Networks provide a way to construct robust patterns from minimally structured information. Certain word disambiguation methods have employed semantic [11, 18, **?**] networks from corpora for more robust similarity measures. Similarly, other models have considered belief propagation networks and Bayesian models for disambiguation. [12] In this research, we consider the degree to which social networks can be used for disambiguation. Recent research has considered a specific case of social networks for unsupervised social disambiguation network [6, 7], in which both ambiguity and variation problems are tackled simultaneously using an iterative approach akin to expectation-maximization. In the maximization step, two references are predicted as the same entity if they are within a certain "distance" of each other. The distance predictions are achieved in the expectation step, and are calculated as a weighed average of 1) the distance between the observed set of references and 2) the groups which the predicted entity for the observed references is expected to be a part of. In the first measure, a measurement between the attributes of the references is incorporate as used in record linkage research (e.g. *John* vs. *Jon*). The second measure corresponds to the distance between two sets of groups, where a group is a clique of entities representative of the document in which the reference resides in, as predicted from the previous iteration.

A shortcoming of this model is a design tailored to an expectation of how citation networks are organized. The proposed model has not been evaluated on actual collaboration networks, but rather synthetic data in which clique structures are guaranteed to exist. As a result, their approach skews predictions towards groups which are not only equivalent, but function as cliques. This bias can have serious difficulty in a lesser connected environment, or decentralized, environments such as the Internet Movie Database studied in this paper. Clique detection requires what we informally term *exact similarity*, such that relationships between entities must be directly observed (e.g. Alice and Bob are related if they collocate in the same source). Furthermore, this model is not necessarily representative of the space of social networks. It is unclear if this model generalizes to other types of social networks [2, 26], such as small-world [22], hierarchical [29], or cellular [10].

As applied in this research, we incorporate community similarity to relax the direct observation requirement and permit relationships to be established be-

tween entities indirectly. For instance, Alice and Bob may never be observed together, but both Alice and Bob collocate Charlie, Dan, and Fran. Though community similarity measures do not necessarily all types of networks, the goal of this research is to demonstrate their capability in comparison to exact similarity in a controlled environment. We suspect that in a less centralized system, such as the IMDB, similarity measures based on community provide more robust metrics. In following section, we introduce two methods: one dependent on exact similarity and an alternative method which is dependent on community similarity.

3 Disambiguation Models and Methods

In this section, we formalize aspects of disambiguation in a more formal manner. In order to do so, we borrow from set theory and introduce a basic set of terminology, definitions, and notations.

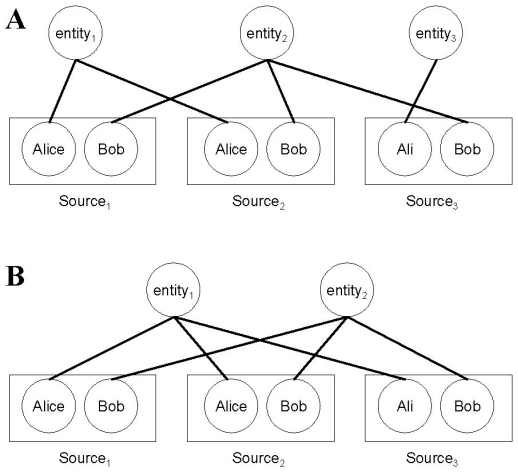


Figure 1: **A**) An example of an ambiguous name *Alice* for $entity_1$ and $entity_3$. **B**) An example of name variation of *Alice* and *Ali* for $entity_1$.

An *entity* is defined as an element from a population of objects. However, entities are not necessarily observed, and thus we consider a set of entities are considered as a set of unobserved, or latent, variables $H = \{h_1, h_2, \ldots, h_k\}$. Rather, there exist a set of objects which are used to reference entities. For this research, we consider these referencing objects to take the form of names. These names manifest in a set of information sources $S = \{s_1, s_2, \ldots, s_m\}$, such that each source s_i consists of a set of extracted names N_i. For example, one can consider a single webpage as a single source. The set of distinct names observed in S is represented by $E = \{e_1, e_2, \ldots, e_n\} = N_1 \cup N_2 \cdots \cup N_m$.

While the same name can be ambiguous to multiple entities, each occurrence of a name references a single entity only. A name which refers to k different entities is called k-ambiguous. This is the scenario depicted in Figure 1.A, where the name *Alice* correctly represents $entity_1$ in $source_1$ and $entity_3$ in $source_3$. Similarly, an entity may be correctly represented by k different names. An entity which is referred to by k different names is called k-variant. In Figure 1.B, $entity_1$ and $entity_2$ are 2- and 1-variant, respectively. For this study, investigation is restricted to 1-variant entities and k-ambiguous names.

In this paper there are two techniques evaluated for name disambiguation, the first leverages directly observed relationships, whereas the second incorporates unobserved, though meaningful, relations. The first technique is a version of hierarchical clustering on sources with ambiguous names only. The second constructs social networks from all sources, regardless of the existence of the ambiguous name of interest. The following sections explain these methods in detail.

3.1 Hierarchical Clustering.

For the first method, each source is represented as a Boolean vector $s_i = [e_{i1}, \ldots, e_{in}]$, where $e_{ij} = 1$ if name e_j is in source s_i and 0 otherwise. Hierarchical clustering is performed using an average linkage criterion calculated as follows. [13] Each source to be clustered is initialized as a singleton cluster. Then, similarity between two clusters c_i, c_j, denoted $csim(c_i, c_j)$, is measured as:

$$csim(c_i, c_j) = \frac{\sum_{s \in c_i, t \in c_j} ssim(s, t)}{|c_i||c_j|}$$

where the similarity between two sources s_i, s_j, denoted $ssim(s_i, s_j)$, can measured using any distance or similarity function. The similarity function of choice for this research is one minus the cosine distance of the vectors of the two source vector representations. More specifically, cosine similarity between two sources is calculated as:

$$ssim(s_i, s_j) = \frac{\sqrt{\sum_{x=1}^{n} e_{ix} e_{jx}}}{\sqrt{\sum_{y=1}^{n} e_{ix}} \sqrt{\sum_{z=1}^{n} e_{iz}}}$$

The most similar clusters are then merged into a new cluster. This process proceeds until either a prespecified stopping criteria is satisfied or all sources reside in one common cluster.

3.2 Random Walks and Network Cuts.

An alternative method considered in this research is the analysis of social networks constructed via names with high certainty. Mainly, we are interested in the partitions of networks as prescribed by random walks from nodes of ambiguous names. One principle difference between the random walk method described in this section and the

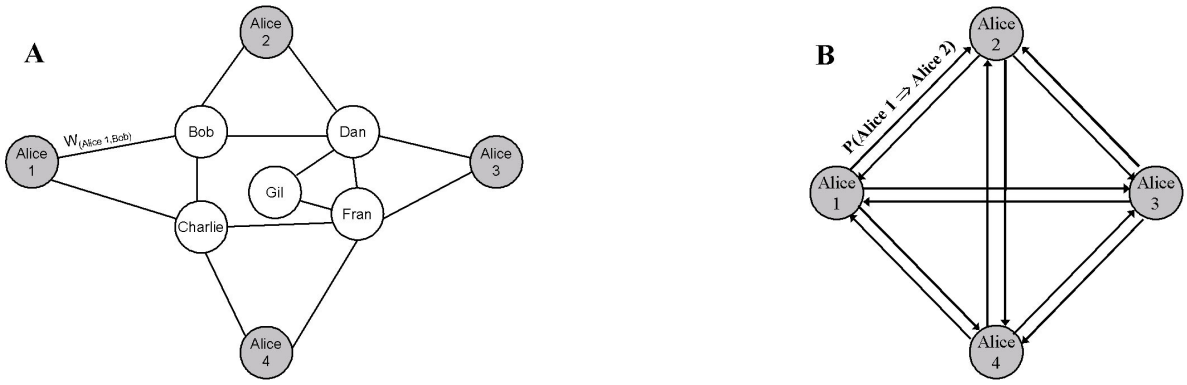


Figure 2: **A**) Social network with a four ambiguous name observations. Nodes connected to ambiguous nodes correspond to original sources. **B**) network with non-ambiguous names removed. The directed edges correspond to the probability of walking from one node to another.

hierarchical clustering of the previous section is the walk is permitted to proceed over nodes (names) which occur in sources devoid of ambiguous names. By doing so, we exploit weak ties, which taken in combination, can permit the discovery of community structures in the graph.

From the set of sources S, a social network is constructed in the following manner. Every distinct name in S is set as a node in the network. An edge exists between two nodes if the names collocate in a source at least one time. The weight of the edge between two nodes i, j is related to the inverse of the number of names observed in a source. This weight is calculated as

$$w_{ij} = \frac{\sum_{s \in S} \theta ijk}{|s|},$$

where θ_{ijk} is an indicator variable with value 1 if names for nodes i and j collocate in source s and 0 otherwise. The reasoning behind this weighting schema is the belief that the lesser number of entities observed in a source, the greater the probability the entities have a strong social interaction. For instance, a website which depicts a list of all students, faculty, and staff of a university conveys less specific information than the class roster for a machine learning graduate course.

In order to test disambiguation in a controlled environment, we make the following adjustment to the networks. For each ambiguous name, we construct a separate network. Basically, the social network is constructed in same manner, except each observation of the ambiguous name of interest is set as its own node in the network. An example network is depicted in image Figure 2.A for the name *Alice*. In this network, *Gil* is indirectly connected to *Alice* through her acquaintances (*Dan* and *Fran*.

Given the social network, we proceed with random walks over the graph. Each walk begins at a node with an ambiguous name observation. The probability a step is taken from node a to node b is the normalized weight of the edge with respect to all edges originating from node a. This probability is calculated as $P(a \rightarrow b|a) = w_{ab} / \sum_j w_{aj}$. Note the probability $P(a \rightarrow a|a) = 0$.

The random walk proceeds from until either 1) an ambiguous name node is encountered or 2) a maximum number of steps are taken. In our studies, we limit the maximum number of steps to 50. After a certain number of random, we approximate the posterior probability of reaching b given the walk originated at a and the observed network, which is represented as $P(a \Rightarrow b)$. As depicted in Figure 2.B, the posterior probabilities remove the necessity for all network nodes except for the ambiguous names. The similarity between nodes a and b is set to the average of the probability of reaching a given b as a start node and vice versa, or $[P(a \Rightarrow b) + P(b \Rightarrow a)]$ / 2. This similarity score is then used in a single linkage clustering process, such that edges are removed if their similarity is below a threshold value. Each resulting components of the graph corresponds to a particular latent variable, or entity. The set of names for each component correspond to the names for a particular entity.

More complex schemes for measuring similarity are proposed in the discussion, but were not evaluated in this study.

3.3 F scores for Multi-class Accuracy. Given a clustering of names, we measuring the accuracy of the predictions through the F-score. This metric was initially introduced by the information retrieval community for testing the accuracy of clusters with greater than two predefined classes, such as the topics of webpages (e.g. baseball vs. football vs. tennis vs. etc..). [23] As applied to disambiguation, the F-score is mea-

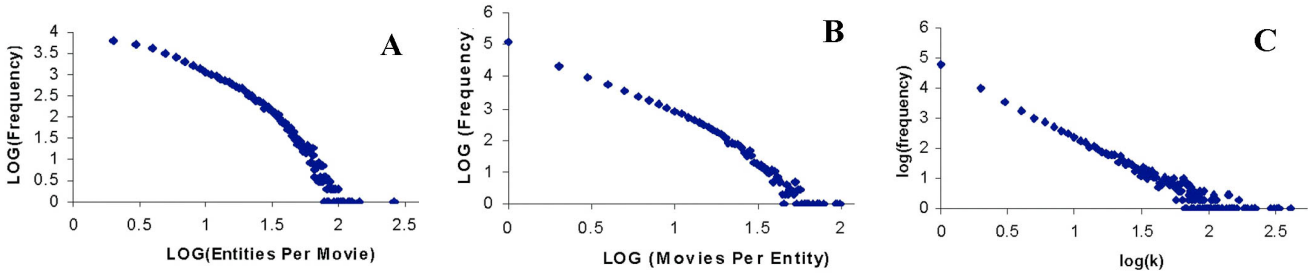


Figure 3: Summary statistics of entity, source, and name distributions in the IMDB. **A**) Log-log plot of movies per entity, **B**) log-log plot of entities per movie, and **C**) log-log plot of frequency of ambiguous name size.

sured as follows. Let $H_e = \{h_1, h_2, \ldots, h_m\}$ be the set of entities referenced by a specific name. Let $S_e = \{s_{e1}, s_{e2}, \ldots, s_{em}\}$ be a set of sets of sources, such that s_{ei} corresponds to the set of sources that entity h_i occurs in. For this research, we only consider sources which contain a single occurrence of an ambiguous name. Thus, for all $s_{ei}, s_{ej} \in S_e, s_{ei} \cap s_{ej} = \emptyset$. Now, let $C = \{c_1, \ldots, c_k\}$ be a set of clusters of the sources in S_e. Furthermore, let $T = \{t_1, \ldots, t_k\}$ be the set of sources for each cluster in C.

The F-score is a performance measure, which uses the harmonic mean of precision and recall statistics for a multi-class classification system. In information retrieval, recall R is defined as the fraction of known relevant documents which were retrieved by the system. In contrast, precision P is defined as the fraction of the retrieved documents which are relevant. For a specific class in the system, which is simply an entity, we define recall and precision for an arbitrary cluster as $R(e_i, c_j) = |s_i \cap t_j|/|s_i|$ and $R(e_i, c_j) = |s_i \cap t_j|/|t_j|$. The F-score for an arbitrary entity-cluster pair, $f(e_i, c_j)$, which is referred to as the local F score, is taken as the harmonic mean of the recall and precision:

$$f(e_i, c_j) = \frac{2R(e_i, c_j P(e_i, c_j)}{R(e_i, c_j + P(e_i, c_j)}$$

While the local F score provides fit for a single entity class and a single cluster, it is the complete system partitioning which we are interested in. To measure the accuracy of the complete system we compute a global F-score, which is basically the sum of the largest local F-scores for each entity class. More specifically, the global F score for an E, C pair is:

$$F(E, C) = \frac{\sum_{s \in S_e} |s| \max_{c \in C}(f(e,c))}{|\bigcup_{s \in S_e} s|}$$

For the methods evaluated in this paper the global F-score is used to test the goodness of fit for a clustering.

4 Experiments

In this section, the disambiguation methods of the previous section are evaluated on a real world dataset.

4.1 Data Description. The dataset chosen to evaluate the disambiguation strategies consists was the Internet Movie Database (IMDB). A publicly available dataset [17] was downloaded from the IMDB's ftp site and was parsed into a relational database for processing purposes. The database contains approximately 115 years worth of actor lists for movies, television shows, straight to video and dvd. For resolution purposes, the IMDB staff labels every entity uniquely, so even entities with ambiguous names are provided with unique primary IDs in the form of an appended roman numeral (i.e. John Doe (I) vs. John Doe (II)). As a result, the underlying truth of the data is known for validation purposes. For this study, this information is only taken into account after disambiguation.

A subset of the IMDB dataset was chosen for evaluation purposes. This subset covered the ten year period 1994-2003 and consists of all movies with greater than 1 actor. For completeness purposes, the following summary statistics were gathered. There are 37,000 movies and 180,000 distinct entities. The distribution of number of movies per actor is depicted in Figure 3.A, and it can be validated that it follows a log-log linear model, or power law distribution. The average number of entities per movie is 8 with a standard deviation of 9.9. Furthermore, it can be validated that in Figure 3.B that the number of entities per movie follows a similar trend. As noted by Barabasi and Albert, the degree distribution of the actor-to-actor network constructed from IMDB data follows a power law distribution as well. [5]

To construct a set of k-ambiguous names, entities were grouped by last name. There are 85,000 distinct last names. The distribution of number of entities per last name also follows a power law distribution,

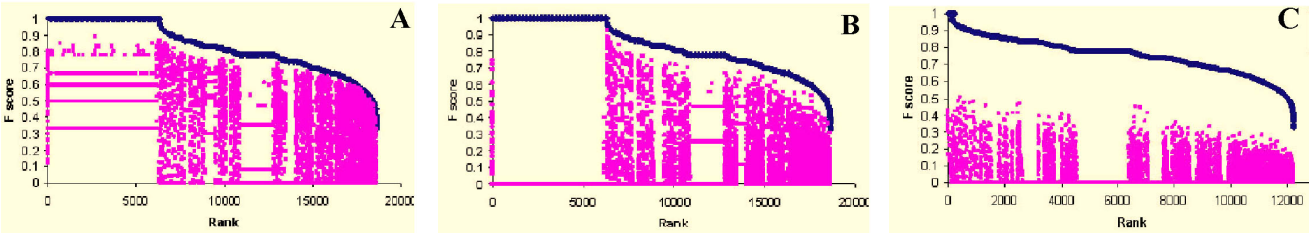


Figure 4: F-scores of hierarchical clustering of sources for each 2-ambiguous name. The topline corresponds to best F score observed during clustering. The plot below is the difference between the best F-score minus a baseline F-score of all sources as **A**) singletons and **B**) a single cluster. Image **C** depicts scores for names where the number of sources is greater than the number names. In this image, the baseline is the difference between the best F-score and the max F-score of both baselines.

as shown in image **C** of Figure 3. To put these numbers in perspective, there are approximately 12,000 2-ambiguous names.

4.2 Hierarchical Clustering Results. The IMDB dataset was subject to hierarchical clustering using the average linkage criteria described above. For clustering raw sources, we considered a continuum of similarity thresholds for stopping the clustering procedure. Figure 4 depicts the best global F-scores achieved for names from this dataset. The x-axis is ordered by number of entities per name, so 1-ambiguous names are on the left. The graph is then subordered by best observed F-score. The predicted F-scores were compared against several baseline methods. In Figure 4.A-C of , the upperline corresponds to the best observed F-score. In Figures 4.A and 4.B, the plot below the best score line corresponds to the difference between the best score and the baseline. The baseline method in Figure 4.A assumes all ambiguous names are distinct entities. In contrast, the baseline in Figure 4.B assumes all ambiguous names correspond to a single entity. These baselines are referred to as *AllSingletons* and *OneClusterOnly*, respectively. In 4, the first 70,000 points correspond to 1-ambiguous names, which explains is why the single cluster baseline predicts perfectly (i.e. F-score of 1).

To consider a more specific case where the baseline is not guaranteed to score perfectly, Figure 4.C depicts a disambiguation results for 2-ambiguous names, where the number of sources is greater than 2. In contrast to Figures 4.A and 4.B, the plot in 4.C presents the difference between the best F-score from hierarchical clustering and the maximum score achievable from a baseline method.

To an extent, the images of Figure 4 skew the clustering prediction results. Though Figure 4 implies that clustering provides F-scores above baseline scores, it must be taken into account that these are the best F-scores possible. The only way to discover the maximum F-score is to check the accuracy of each disambiguation prediction against the underlying truthful values. It is unfair to compare the power of hierarchal clustering to maximum F score of the baseline tests for similar reasons. Just as we cannot consider all partitions of the hierarchical clustering process simultaneously, we cannot simply take the max of both baselines - we must choose one or the other. In reality, an automated method must be able to find a point at which clustering automatically stops.

A simple method which was tested for automatic stopping was to average out the F-scores at various similarity threshold values. The resulting scores are demonstrated in Figure 5.A with the label "hc". In contrast to Figure 4, the average F-scores for all singletons and single cluster baselines are reported. The vertical line in the graph depicts one standard deviation around the average hierarchical clustering F-score. A threshold of 0 corresponds to the *OneClusterOnly* baseline and a threshold of 1 corresponds to the *AllSingletons* baseline. In Figure 4.A, as the threshold increases from 0 to 1, the F-score increases. The average F-score reaches a maximum value close to a similarity of 0.99, at which point the average F-score and all clusterings within 1 standard deviation achieve better than the best baseline of all singletons. This is very encouraging, except with such a high similarity threshold it is implied that we should only merge clusters with extremely high structural equivalence in their vectors. This is quite peculiar, and appears to be completely antithetical to the belief that community structures permit greater capability for disambiguation.

4.3 Random Walk Results. However, once we consider the results from the random walk clustering, the previous result appears to be less counter than initially implied. In the right plot of Figure 4, we present average

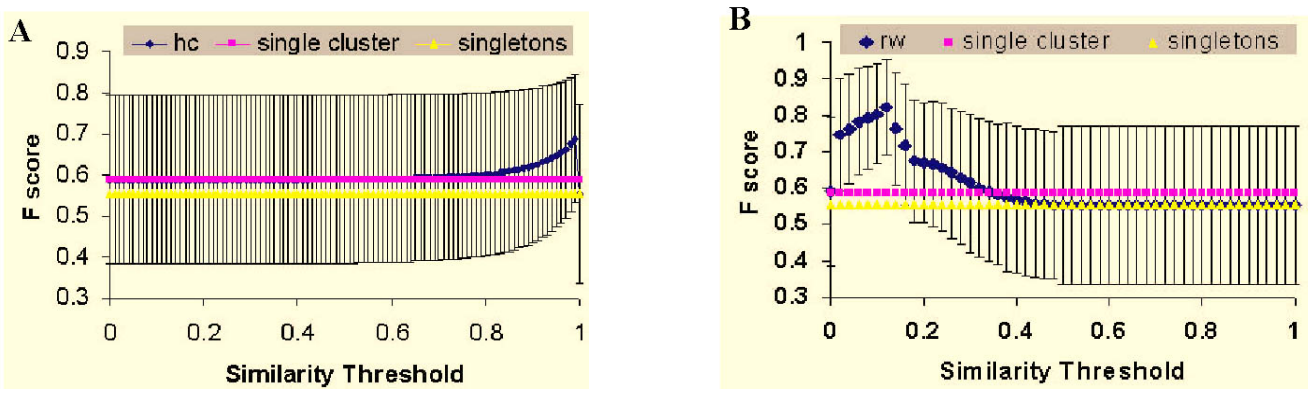


Figure 5: **A**) Average F-score of hierarchical clustering (hc), singletons, and single cluster baselines over continuum of cosine similarity threshold values. The vertical lines correspond to 1 standard deviation. **B**) Average F-score of random walk network partitioning, singletons, and single cluster baselines over continuum of cosine similarity threshold values. The vertical lines correspond to 1 standard deviation.

the F scores for random walk partitioning. There were 100 random walks initiated from each ambiguous node. Recall, similarity is actually the mean of the probability of walking between ambiguous name observations a and b within 50 steps. The graph is then thresholded, such that probabilities below the threshold are removed, and the resulting network components are set as the predicted clusters. From this plot, it is apparent that a maximum F-score is achieved at a relatively low threshold, specifically a probability of 0.12. Moreover, the average F score maximum at this point is greater that the maximum for simple hierarchical clustering by approximately 0.1. This is a significant improvement and supports the community structure hypothesis. Nodes and edges which are not directly related to the ambiguous names provide a significant amount of power for disambiguation purposes.

5 Discussion

The results of the previous section demonstrate community equivalence provides an advantage over exact equivalence for measuring similarity and, subsequently, disambiguation. While the datasets which these results are derived correspond to real world observations, the experiments and models of disambiguation are based on a highly controlled environment. Some of the limitations of this environment, and possibilities for extension are addressed in the following sections.

5.1 Building a Better Stopping Criteria. One limitation of this work stems from its dependency on a static threshold as a stopping criteria of the clustering process. This is a age old concern regarding hierarchical clustering and, for the most part, all stopping criteria are based on heuristics which are tailored to a researcher's respective environment. Airoldi and Malin have recently proposed a statistical test for stopping the clustering process based on geometric intuition regarding the growth rates of clusters. [1] In their research, clustering utilizes a single linkage criterion and thus has yet to be proven if such geometric insights hold for more complex clustering criteria such as the average linkage method employed for this paper's analysis. It is possible such tests could be adapted and in future research we hope to address this issue in more depth.

Though stopping criteria for hierarchical clustering may be difficult to define, it might be easier to derive an intuitive threshold for the random walk procedure. In this research, only similarity based on the probability of reaching one node from another was considered. However, this is an incomplete picture of the community surrounding an ambiguous name, and furthermore is a biased estimator. The information which random walks provide is much more substantial than the probability of reaching one node from another. In effect, there are at several additional features which can be accounted for to reduce bias in static thresholds. First of all, certain names are observed in more sources than other names. As a result, if the probability of reaching node b from node a is 0.2 and there are 20 sources in consideration, this is clearly a more probable occurrence than if the same probability was observed when 200 sources are considered.

Second, random walks provide the probability that a node will reach any node. Thus, we can consider the number of times a walk originating from an ambiguous node finds another ambiguous node, including itself, in the random walk. Note, there will be occurrences

when a random walk fails to find an ambiguous nodes. Such occurrences should not be discounted since they still communicate important indications of the distance between one ambiguous node and another. Thus, it is apparent that the probability $P(a \Rightarrow b)$ should be inversely correlated to the probability a node walks back to itself, or $P(a \Rightarrow a)$. Furthermore, we should negatively reweight if node b is a node which is reachable from many different nodes.

Third, the random walks were arbitrary specified to time out after 50 steps. By this construction, a walk completed successfully (i.e. reaches an ambiguous name node) in 2 steps is given equal weight in the similarity measure than a successful walk of 50 steps. It is possible that a discounting model may be more appropriate, such that as the number of steps increases, the score provided to a successful completion tends toward zero. In future research we expect to design more formal probabilistic representations of community similarity.

5.2 Towards More Realistic Models. In this paper, we introduced the concept of a k-ambiguous name. While there were almost 20,000 names with a k greater than 1, we controlled our clustering experiments to test on environments where the only uncertainty was associated with one particular name. Controlling for certainty is useful in the evaluation of the relative performance of disparate disambiguation procedures, but obviously this is an unrealistic assumption. In the real world, it is not clear if any observed name ever has complete certainty. This suggests that probabilistic models of certainty may be useful for disambiguating names when many names are ambiguous. For instance, expectation-maximization strategies over the graph are a potential route of research for resolution. [20, 21] With respect to this research, an extension to this research is to consider basic iterative methods, which can be used to cluster and classify relational data by leveraging names of high certainty, which can be fixed, or removed, during the learning process. By doing so, we can take advantage of high certainty knowledge to resolve lesser certain situations. We intend to investigate such models in future research.

Furthermore, as noted in previous works [5], the IMDB actor-to-actor network is variant of a random network with strong clustering features. In order to test disambiguation on a larger scale, we expect to test our methods on other types of social networks.

5.3 Making Search Engines More Social. Though there are limitations to the disambiguation research set forth in this paper, the results are promising and there exist potential applications for the next generation of search engines. This is especially so for search engines which archive and retrieve documents with large numbers of names. Clustering webpages based on their disambiguation properties can assist in making retrieval responses to queries more meaningful. Rather than rank pages by relevance using methods based on spectral decomposition properties, which are simply bag of words similarity comparisons, pages of relevance could be partitioned into clusters regarding the particular entities of interest. When results are displayed to the user, each ambiguous name could be qualified by key words extracted from the documents in the cluster. Obviously, this is speculation into an approach for search engines; nonetheless, the methods evaluated in this paper can provide a basis for future research and development of socially cognizant search engines.

6 Conclusions

This paper evaluated several methods for disambiguating names in a relational environment (actor collaborations in the Internet Movie Database) were presented. The first method was based on hierarchical clustering of sources in which ambiguous names are observed. The second method leveraged social networks constructed from all sources, such that random walks originating from ambiguous name nodes, were used to estimate posterior distributions of relations to partition the graph into components. We controlled social networks to study a single ambiguous name, and our findings suggest methods which leverage community, in contrast to exact, similarity provide more robust disambiguation capability. This research served as proof of concept for social network-based disambiguation, and in the future we will generalize our methods to account for networks that consist of more than one ambiguous names.

References

[1] E. Airoldi and B. Malin. Data mining challenges for electronic safety: the case of fraudulent intent detection in e-mails. In *Proc IEEE ICDM-2004 Workshop on Privacy and Security Aspects of Data Mining*. Brighton, England. 2004.

[2] R. Albert and A.L. Barabasi. Statistical mechanics of complex networks. *Reviews of Modern Physics*. 2002; 74: 47-97.

[3] A. Bagga and B. Baldwin. Entity-based cross-document coreferencing using the vector space model. In *Proc* 36^{th} *Annual Meeting of the Association for Computational Linguistics*. San Francisco, CA. 1998; 79-85.

[4] M. Banko and E. Brill. Scaling to very large corpora for natural language disambiguation. In *Proc* 39^{th}

Annual Meeting of the Association for Computational Linguistics. Toulouse, France. 2001.

[5] A.L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*. 1999; 286: 509-512.

[6] I. Bhattacharya and L. Getoor. Iterative record linkage for cleaning and integration. In *Proc* 9^{th} *ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*. Paris, France. 2004; 11-18.

[7] I. Bhattacharya and L. Getoor. Deduplication and group detection using links. In *Proc 2004 ACM SIGKDD Workshop on Link Analysis and Group Detection*. Seattle, WA. 2004.

[8] E. Brill and P. Resnick. A rule-based approach to prepositional phrase attachment disambiguation. In *Proc* 15^{th} *International Conference on Computational Linguistics*. Kyoto, Japan. 1994; 1198-1204.

[9] P. Brown, S. Della Pietra, V. Della Pietra, R. Mercer. Word-sense disambiguation using statistical methods. In *Proc* 29^{th} *Annual Meeting of the Association for Computational Linguistics*. Berkeley, CA. 1991; 264-270.

[10] K.M. Carley, M. Dombroski, M. Tsvetovat, J. Reminga, and N. Kamneva. Destabilizing dynamic covert networks. In *Proc* 8^{th} *International Command and Control Research and Technology Symposium*. Washington, DC. 2000.

[11] S. Chan and J. Franklin. Symbolic connectionism in natural language disambiguation. *IEEE Transactions on Neural Networks*. 1998; 9(5): 739-755.

[12] G. Chao and M.G. Dyer. Word sense disambiguation of adjectives using probabilistic networks. In *Proc* 17^{th} *International Conference on Computational Linguistics*. Saarbrucken, Germany. 2000; 152-158.

[13] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification, 2nd Edition*. Wiley. New York. 2001.

[14] W.A. Gale, K.W. Church, and D. Yarowsky. A method for disambiguating word senses in large corpora. *Computers and Humanities*. 1992; 26: 415-439.

[15] F. Ginter, J. Boberg, J. Jarvinen, and T. Salakoski. New techniques for disambiguating in natural language and their application to biological text. *Journal of Machine Learning Research*. 2004; 5: 605-621.

[16] V. Hatzivassiloglou, P.A. Duboue, and A. Rzhetsky. Disambiguating proteins, genes, and RNA in text: A machine learning approach. *Bioinformatics*. 2001; 17:97-106.

[17] The Internet Movie Database. http://www.imdb.com.

[18] K. Hiro, H. Wu, and T. Furugori. Word-sense disambiguation with a corpus-based semantic network. *Journal of Quantitative Linguistics*. 1996; 3: 244-251.

[19] K. Jensen and J.L. Binot. Disambiguating prepositional phrase attachments by using on-line definitions. *Computational Linguistics*. 1987; 13(3-4): 251-260.

[20] D. Jensen and J. Neville. Iterative classification of relational data. *Papers of the AAAI-2000 Workshop on Learning Statistical Models From Relational Data*. AAAI Press. 2000.

[21] D. V. Kalashnikov and S. Mehrotra. A probabilistic model for entity disambiguation using relations. *Computer Science Department Technical Report TR-RESCUE-04-12*, University of California, Irvine. June 2004.

[22] J. Klienberg. The small-world phenomenon: An algorithmic perspective. In *Proc* 32^{nd} *Annual ACM Symposium on Theory of Computing*. Portland, OR. 2000.

[23] B. Larsen and C. Aone. Fast and effective text mining using linear-time document clustering. In *Proc* 5^{th} *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. San Diego, CA. 1999; 16-22.

[24] M. Lesk. Automatic sense disambiguation: How to tell a pine cone from an ice cream cone. In *Proc* 1986 *ACM SIGDOC Conference*. New York, NY. 1986; 24-26.

[25] G. Mann and D. Yarowsky. Unsupervised personal name disambiguation. In *Proc* 7^{th} *Conference on Computational Natural Language Learning*. Edmonton, Canada. 2003.

[26] M. Newman. The structure and function of complex networks. *SIAM Review*. 2003; 45, 167-256.

[27] H.T. Ng. Exemplar-based word sense disambiguation: Some recent improvements. In *Proc* 2^{nd} *Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics. Somerset, New Jersey. 1997; 208-213.

[28] H.T. Ng, B. Wang, and Y.S. Chan. Exploiting parallel texts for word sense disambiguation: an empirical study. In *Proc* 41^{nd} *Annual Meeting of the Association for Computational Linguistics*. Sapporo, Japan. 2003.

[29] E. Ravasz and A.L. Barabasi. Hierarchical organization in complex networks. *Phys. Rev. E.* 2003 ; 67, 026112.

[30] L. Sweeney. Finding lists of people on the Web. *ACM Computers and Society*. 2004; 34(1).

[31] V. Torvik, M. Weeber, D.W. Swanson, and N.R. Smalheiser. A probabilistic similarity metric for medline records: a model of author name disambiguation. *Journal of the American Society for Information Science and Technology*. 2004; 55(13): forthcoming.

[32] J. Vronis and N. Ide. Word sense disambiguation with very large neural networks extracted from machine readable dictionaries. In *Proc* 13^{th} *International Conference on Computational Linguistics*. Helsinki, Finland. 1999; 389-394.

[33] N. Wacholder, Y. Ravin, and M. Coi. Disambiguation of Proper Names in Text. In *Proc* 5^{th} *Applied Natural Language Processing Conference*. Washington, DC. 1997.

[34] D. Yarowsky. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. In *Proc* 30^{th} *Annual Meeting of the Association for Computational Linguistics*. Nantes, France. 1992; 454-460.