# Stony Brook University

# Analyzing Dynamics in Online Social Networks

A Dissertation Presented
by
**Akshay Patil**

to
The Graduate School
in Partial Fulfillment of the
Requirements
for the Degree of
**Doctor of Philosophy**
in
**Computer Science**
Stony Brook University

**August 2013**

**Stony Brook University**
The Graduate School

**Akshay Patil**

We, the dissertation committee for the above candidate for the
Doctor of Philosophy degree, hereby recommend
acceptance of this dissertation.

**Jie Gao – Dissertation Advisor**
Associate Professor, Department of Computer Science

**Steven Skiena – Chairperson of Defense**
Professor, Department of Computer Science

**Arnout van de Rijt**
Assistant Professor, Department of Sociology

**Juan Liu – External Member**
Senior Research Scientist, Palo Alto Research Center Inc.

This dissertation is accepted by the Graduate School

Charles Taber
Interim Dean of the Graduate School

Abstract of the Dissertation

**Analyzing Dynamics in Online Social Networks**

by

**Akshay Patil**

**Doctor of Philosophy**

in

**Computer Science**

Stony Brook University

**2013**

As the complexity of online activities increases, social network structures have come to play an increasingly important role in the experience and effectiveness of an individual's online life. These structures exhibit a high degree of dynamism. Also, online platforms have provided us with unprecedented opportunities to study behavior and dynamics of these network structures. Understanding whether/why/when a person will behave in a certain manner can be important in a number of social domains. In our work, we model these network dynamics and design accurate prediction algorithms for these behavioral models.

The main challenges that we address in this dissertation are 1) the ability to predict imminent departure events and the probable adverse impact of these events, 2) understanding the processes that drive group growth and stability, and 3) implications of social influence on opinion and relationship formation. Our work offers interesting insights on factors that cause and affect these network events. The methods proposed in our study use a diverse set of features that help us in building richer predictive models that result in more accurate predictions. Another aspect of our research deals with the innovative use of spectral graph theory concepts in unifying activity information of people across different social platforms. We also contribute in the development of a large-scale news and blog analysis engine that provides ready access to a wealth of interesting statistics on millions of people, places, and things across a number of interesting web corpora.

The work we present has a wide range of applications: helping spot malicious behavior, forecasting group stability, predicting churn, recommending better content, deanonymizing network identities, and detecting trends in news data. Our techniques have been evaluated and validated on several large-scale, real-world datasets that span different domains.

*Dedicated to my grandparents.*

# Contents

# List of Tables

# List of Figures

# Acknowledgements

Over my six years of graduate study, I have had the privilege to work & interact with a number of remarkable people who have made my time at Stony Brook University enjoyable and rewarding. I would like to thank all of them; without them this dissertation would not have been possible.

I begin by expressing my heartfelt thanks to my advisor Professor Jie Gao. It is through Jie that I came upon the fascinating field of social network research. I believe that our foray into this field has been a great learning experience. Her support, guidance & infinite patience has been instrumental in bringing this work to fruition. Her eagerness to learn, humble nature & professionalism is exemplary; facets that have made our collaboration a pleasurable journey.

Further, I owe my gratitude to Juan Liu for mentoring me on the ADAMS project throughout the past couple of years. Her ability to come up with ideas that are simple, yet effective is truly remarkable. A large chunk of this dissertation & some of my best work is the result of my collaboration with Juan. I also owe to her, two wonderful summers that I spent interning at PARC, which allowed me to interact and work with some remarkable people.

I also cherish the time I spent working on the Lydia project with Professor Steven Skiena and his research group. I would like to thank Steve for welcoming me to his research group. Steve is the quintessential cool guy in the department! He is also an excellent taskmaster and a very nice person to talk to. I thank him for answering all the crazy questions I have had throughout my graduate life.

I also value the research collaboration and input provided by Professor Arnout van de Rijt. He has greatly helped me in understanding the theories behind social networks and their evolution.

I would also like to thank all my co-authors, especially those who have made

contributions to this dissertation: Arnout van de Rijt, Bob Price, Charles Ward, Golnaz Ghasemiesfeh, Hossam Sharara, Jianqiang Shen, Jie Gao, John Hanley, Juan Liu, Leonidas Guibas, Mikhail Bautin, Nan Hu, Oliver Brdiczka, Roozbeh Ebrahimi & Steven Skiena.

Special thanks to all my colleagues in the Algorithms Lab. They have helped liven up the atmosphere in the lab. In particular, I would like to acknowledge Anurag Ambekar, Charles Ward, Dmytro Molkov, Dzejla Medjedovic, Girish Kathalagiri, Golnaz Ghasemiesfeh, Jiemin Zeng, Mayank Goswami, Mikhail Bautin, Pablo Montes Arango, Roozbeh Ebrahimi, Sam McCauley, Shashank Naik & Shrijeet Paliwal.

I owe a token of appreciation to the administrative staff at Stony Brook University, especially to Betty Knittweis, Brian Tria, Christos Kalesis, Cynthia Scalzo & Jasmina Gradistanac.

On the personal front, I would like to thank my girlfriend Akshata, my parents & my grandparents for their endless love and understanding. It would have been impossible to complete this dissertation without their continued support and encouragement. A large amount of credit is also owed to Girishkumar Sabhnani. Girish is the best friend that everyone wants, but few get! His never-ending belief in me has helped me throughout this journey.

Lastly, I have been fortunate enough to have some wonderful friends. Thanks to Abhishek, Akash, Ameya, Anupam, Manas, Omkar, Pankaj, Pralhad, Radhika, Rajul, Rushang, Sanket, Saurabh, Sujan & Swapnil for all the wonderful times together.

# Chapter 1

# Introduction

A *social network* is a graph structure that consists of a set of *social entities* and a set of *relationships* between these entities. Study of these structures towards understanding their evolution, modeling the network dynamics, and being able to identify local and global patterns is what constitutes the field of *Social Network Analysis* (SNA). Social Network Analysis is a remarkably interdicisplinary field consisting of contributions from sociology, psychology, computer science, statistics & mathematics.

The proliferation of online activity and social networking platforms have provided the research community with unprecedented opportunities to study and analyze the social network structures on a massive scale. Questions such as, "How does the network graph look like? How will the social network evolve? How can we identify influential nodes in networks? Can we predict your friends?" are now being answered by applying social network analysis techniques to large-scale online data. Our study extends the frontier by helping answer the following questions,

- Can we predict departure events in online social networks? Also, can we quantify and predict the possible impact of these departure events?
- Can we identify the factors that lead to instability in groups in networks? Also, can we predict whether a group is likely to remain stable or shrink over a period of time?
- Do our friends or foes influence us in forming new relationships or in forming opinions about others?

Apart from helping us analyze network dynamics, our study also builds techniques that help in unifying activity information of people across different social platforms or across two temporally seperated snapshots of the same platform. Finally, our work is also beneficial in uncovering local and global patterns in news and blog data, that helps in a better understanding of the political, social, and cultural world around us.

## 1.1 Preliminaries

In this section, we briefly go over basic concepts that will be used throughout this dissertation. A social network can be modeled as a graph $G(\mathcal{V}, \mathcal{E})$ where the social entities are represented by nodes ($v \in \mathcal{V}$) and the relationships between these entities are represented by edges ($e \in \mathcal{E}$). Figure 1 provides a visualization of the collaboration network for Akshay Patil. In this network, we have one type of entity (researcher) and one type of relationship (collaboration) between these entities. Also, the network is a static snapshot i.e. there is no notion of representing time in the network. In our study, we deal with complex, dynamic network structures that vary over time and include multiple types of entities & relationships.

**Adjacency Matrix:** A graph $G(\mathcal{V}, \mathcal{E})$ can be represented by a $N \times N$ ($N = |\mathcal{V}|$) matrix $A$, where $A_{i,j} = 1$ if $(i, j) \in \mathcal{E}$ and 0 otherwise.

**Laplacian Matrix:** A graph can also be represented by the *Laplacian Matrix* $\mathcal{L} = D - A$, where $A$ refers to the *adjacency matrix* and $D$ is the *diagonal degree matrix* ($D_{ii} = \sum_j A_{ij}$).

**Undirected & Directed Graph:** A graph is said to be an *undirected graph* if the edges have no orientation. The edge $(i, j)$ is identical to the edge $(j, i)$. On the other hand, if the edges have direction, then the graph is a *directed* graph.

**Node Degree:** The degree of a node is given by the number of edges connected to it. For directed graphs, there are two notions of node degree. *Out-degree* $d_{out}$ denotes the number of edges pointing from the node & *in-degree* $d_{in}$ denotes the number of edges pointing towards the node. For an undirected graph, $d_{in} = d_{out}$.

**Connected Component:** A *connected component* of an undirected graph is a maximal set of nodes where any two nodes in the component are connected to each other by paths.

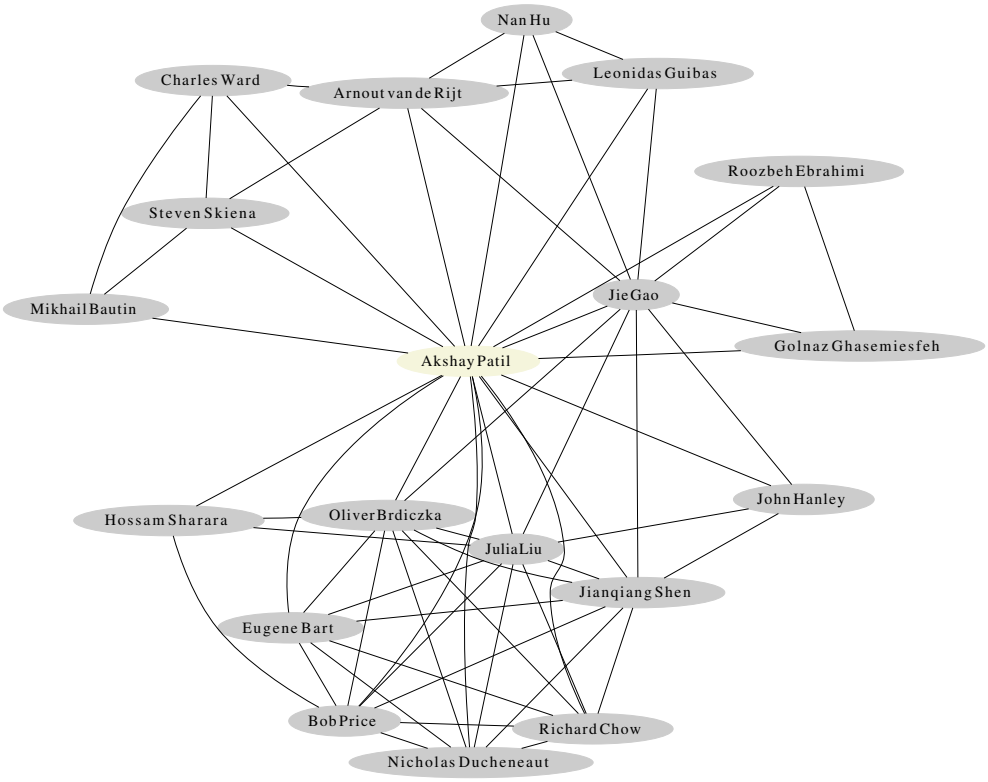



Figure 1: Collaboration Network for Akshay Patil. The nodes represent researchers and an edge is placed between two researchers that have collaborated on a project or a paper.

**Complete Graph:** A *complete graph* is a graph in which every pair of nodes is connected by an edge. A complete undirected graph of $n$ nodes has $\frac{n(n-1)}{2}$ edges.

**Triangle:** A *triangle* is a tuple of 3 connected nodes $(i, j, k)$ such that $(i, j), (j, k), (i, k) \in \mathcal{E}$.

**Dynamic Graph:** A *dynamic graph* is one that varies its structure over time. Each entity & relationship of a dynamic graph is associated with a temporal variable that

records the creation time of the corresponding entity or relationship. A dynamic graph $G_t(\mathcal{V}_t, \mathcal{E}_t)$ represents the graph as it exists at time $t$.

**Multi-Modal, Multi-Relational Graph:** A *multi-modal, multi-relational graph* refers to a graph that consists of multiple types of entities and allows for different kinds of relationships across the multiple entities. Such a graph can be represented by $G(\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = (V^1, V^2 \dots V^k)$ represents the different types of entities/nodes and $\mathcal{E} = (E^1, E^2 \dots E^l)$ represents the different types of relationships.

**Clustering Coefficient:** *Clustering coefficient* is a topological concept, measuring the degree to which nodes in a graph tends to be clustered together. Formally it is measured as the ratio between the number of closed triplets over the number of connected triples of vertices. It takes value 0 in a star topology and value 1 if the graph is fully connected. For example, if a node $v$ has $d$ neighbors, then atmost $d(d-1)/2$ edges can exist between them. The clustering coefficient for node $v$ denotes the fraction of these allowable edges that actually exist. The *global clustering coefficient* is then defined as the average of the local clustering coefficients of all the nodes in the graph.

**Degree Distribution:** The degree of a node gives the number of connections this node has to other nodes in the graph. The *degree distribution* is the probability distribution of degrees of all the nodes in the network. The degree distribution of a graph follows a *power-law* distribution if the number of nodes $N_d$ of degree $d$ is given by $N_d = d^{-\gamma}$. $\gamma > 1$ and is called the *power law degree exponent*. Such a distribution indicates that majority of nodes have low degree but a small number of them have high degree. These high degree nodes are often called "hubs" of the network. Many real-world social networks are found to have degree distributions that approximately follow a power-law.

## 1.2 Overview

The dissertation can be divided into four parts. The first part includes a problem of modeling & predicting departure events in network data, followed by its application

to predicting group stability in online social networks. The second part of the dissertation deals with studying social bias that may exist in consumer review websites. In the third part, we consider the identity management problem of matching two anonymous networks that share a majority of their nodes. Lastly, the fourth part of the dissertation introduces the TextMap Access system, which provides ready access to a wealth of interesting statistics on millions of people, places, and things across a number of interesting web corpora.

The first part starts from a fundamental problem of modeling group dynamics. Social groups often exhibit a high degree of dynamism. Some groups thrive, while many others die over time. Modeling group dynamics and understanding whether/when a group will remain stable or shrink over time can be important in a number of social domains. Specifically, we look at the following problems in this area. In the first problem, we build models to predict if and when an individual is going to quit his/her group, and whether this quitting event will inflict substantial damage to the group. We take the World of Warcraft game as an exemplar platform for studying this problem. Our proposed solution starts from in-game census data and extracts features from multiple perspectives such as individual-level, guild-level, game activity, and social interaction features. These features are then used to build predictors. Our study shows that destructive group dynamics can often be predicted with modest to high accuracy, and feature diversity is critical to prediction performance. A related problem that we tackle looks at churn/attrition within an organization and tries to model this churn. Our approach provides us with valuable insights into understanding attrition within an organization. Lastly, we extend our analysis to be able to predict group stability. We build models to predict if a group is going to remain stable or is likely to shrink over a period of time. Results indicate that both the level of member diversity and social activities are critical in maintaining the stability of groups. We also find that certain 'prolific' members play a more important role in maintaining group stability.

The second part of the dissertation studies social influence on consumer review websites. Many eCommerce websites and consumer review websites provide both reviews of products and a social network structure among the reviewers. Users can review products they purchased as well as the reviews others wrote. Users can also rate each other as trusted or untrusted relationships. By studying a data set from

Epinions, we examine and quantify how the trust/distrust relationships among the users influence their ratings of the reviews. We discover that the opinions of one's friends has an influence on his/her ratings, while the opinions of one's foes seem to have no influence. We also find that a user's friends play a significant role in guiding his/her future relationships.

The third part of this dissertation focuses on the identity management problem of matching two anonymous networks that share a majority of their nodes. We look at this problem from the viewpoint of social networks. Specifically, we aim to identify the nodes in two social networking graphs such that the graph topologies are mostly aligned. The problem is a variant of the well-studied subgraph isomorphism problem and there are a range of potential applications in social network analysis, from assessing the possibility for unification of social activity information of users across different online social platforms to detecting who is who in two networks that capture communication patterns of terrorists. The solution we propose is a family of informative graph signatures on each node, defined as the Laplacian Family Signatures (LFS), that describes the role of a node relative to other nodes in the graph topology. The LFS is naturally multi-scale and captures global network structural information beyond immediate neighborhoods. By matching nodes in two graphs with similar LFS, we develop a practical algorithm to identify nodes in two graphs with sim- ilar topology. Analytically we also demonstrate the continuity of LFS under small graph perturbations for different noise models. Through comparison of signatures within the family and over some other state-of-the-art candidate methods on both randomly generated graphs and real world social networks, we show the outperformance of LFS.

The last part of the dissertation details the TextMap Access system, which provides ready access to a wealth of interesting statistics on millions of people, places, and things across a number of interesting web corpora. The system is powered by a flexible and scalable distributed statistics computation framework using Hadoop. The continually updated corpora includes newspapers, blogs, patent records, legal documents, and scientific abstracts; well over a terabyte of raw text and growing daily. Our work on the web-user interface provides instant access to the system for students and scholars. Our contribution is a valuable addition to the social scientist's toolbox.

## 1.3 References

The work included in this dissertation has been or will be published in international conferences and journals, and is thus under copyright. The list of references is as follows,

Chapter 2 is the result of collaboration with Juan Liu, Bob Price, Hossam Sharara & Oliver Brdiczka. A paper titled "Modeling Destructive Group Dynamics in On-Line Gaming Communities" has been published in the *Proceedings of the 6th International AAAI Conference on WebBlogs and Social Media (ICWSM 2012)* which was held in Dublin, Ireland from June 4-8, 2012 [114]. Part of the work was also presented at the *Workshop on Research for Insider Threat (WRIT 2012)* that was conducted in San Francisco, CA on May 24-25, 2012 [23].

Chapter 3 results from the colloboration with Juan Liu & Jie Gao. A paper titled "Predicting Group Stability in Online Social Networks" has appeared in the *Proceedings of the 22nd International Conference on World Wide Web (WWW 2013)*. The conference was held in Rio de Janeiro, Brazil from May 13-17, 2013 [113].

Chapter 4 details recent work performed with Juan Liu, Jianqiang Shen, Oliver Brdiczka, Jie Gao & John Hanley. A paper titled "Modeling Attrition in Organizations From Email Communication" has been submitted to the *ASE/IEEE International Conference on Social Computing (SocialCom 2013)* [115].

Chapter 5 is the recent work coauthored with Golnaz Ghasemiesfeh, Roozbeh Ebrahimi & Jie Gao. A paper titled "Social Influence in Epinions: A Case Study" has been submitted to the *ASE/IEEE International Conference on Social Computing (SocialCom 2013)* [112].

Chapter 6 is the result of collaboration with Nan Hu, Jie Gao, Arnout van de Rijt & Leonidas Guibas [66].

Chapter 7 is the work coauthored with Mikhail Bautin, Charles Ward & Steven Skiena. A paper titled "Access: News and Blog Analysis for the Social Sciences" has been published in the *Proceedings of the 19th International Conference on World Wide Web (WWW 2010)*. The conference was held in Raleigh, NC from April 26-30, 2010 [15].

# Chapter 2

# Modeling Destructive Dynamics in Online Communities

## 2.1 Introduction

As the complexity of online activity has increased, formal group structures have come to play an increasingly important role in the experience and effectiveness of an individual's online life. While formal groups can be seen in many task-oriented online communities, the use and impact of formal groups in online role playing games is among the most highly developed at this time. In these games, players join groups known as guilds, some of which include hundreds of members, to share resources, plan strategy and execute large scale attacks on opposing forces. The effectiveness of these groups can be undermined when group members depart, taking with them, experience, resources and possibly other group members. The ability to predict imminent departure of group members and the probable impact of their departure is highly desirable, as it offers insights on factors that affect online group effectiveness. It also provides practical guidance to tasks such as risk management and customer retention.

Instability in on-line role-playing game groups has been well documented [44, 45]. Specific factors identified as contributing to member departures include guild leadership style [146], disengagement from the online community itself [149], and internal conflicts between members. Most previous works [44] [146] [149] have

investigated guild membership dynamics using qualitative case studies and small sample surveys of selected players. These studies have provided us with many qualitative insights, but have not yet reached the level of practical mathematical predictors that can be deployed online to make predictions based on actual game data.

Automated analysis has been applied to the discovery of networks that support undesirable activities such as gold farming [80] [62] and the trade of contraband items [4]. Network analysis has also be used to investigate trust among players [5]. We also exploit properties of the player's interaction with the community to make predictions, however we are unaware of prior work specifically addressing the prediction of departures from groups and the estimation of associated damage. Perhaps the closest work to ours is the research on churn prediction in MMOGs [20]. Player churn occurs when players stop playing the game (a more extreme event than switching guilds). The authors employ machine learning classifiers directly on game data to predict game departure. In churn prediction, the player's overall game satisfaction (as measured by achievement) is the key driver. In contrast, after a guild departure, the player is often still highly engaged in the game and continues play with a different guild. The motivational structure is therefore quite different. As a consequence, the features for prediction are also different. Churn detection primarily rely on game achievement features, while our guild quitting prediction weighs heavily on features regarding the quality of social interactions within a specific guild.

In addition to identifying players likely to quit their guild, we also wish to predict when the quitting event will happen. Models such as the dynamic influence model [9, 40, 109] have been used to represent changing patterns of social interactions. This Bayesian model uses a detailed description of the conditional dependence between each player's current state at time $t$ and the previous states of all players at time $t-1$. While powerful, the model is computationally expensive, and requires detailed modeling of social interactions between actors making it difficult to scale to large networks.

The chapter is organized as follows. It starts with an overview of WoW data and guild-quitting statistics, followed by a brief investigation of whether social interaction plays any role in guild-quitting decisions. We then move on to define the

impact of an imminent quitting event. The later sections present predictive models to predict (1) the potential impact and (2) if and when a quitting event will happen. We conclude with a discussion of the important factors for predictions and an outline of future work.

Though the work presented in this chapter is focused on the specific problem of guild quitting dynamics in WoW, we hope that similar ideas and approaches can be generalized to other social domains. Group destruction is a common phenomenon, for instance, an employee quitting job in a corporate environment may often have co-workers following him/her to join a new company. Likewise, market studies have shown that customer loyalty to a product can be weakened if the customer's close friend opt out to a new product. It is desirable to gain the insights on social group dynamics and the capability of prediction will be highly valuable.

## 2.2 Overview of WoW Data and Guild-quitting Events

To explore guild quitting dynamics, we use data from a previous WoW study [45]. A web-based crawler was deployed to log in-game activities based on the API specified by Blizzard Entertainment, the producer of WoW. The crawler periodically issues "/who" requests every 5 to 15 minutes, depending on server load, to get a list of characters currently being played on a given server. Over six months of data are logged, from November 2010 to May 2011. The data is sometimes referred to as the WoW census. Three types of servers are logged: player-vs-environment (PvE), player-vs-player (PvP), and role playing (RP). The servers may present players with different game tasks, but are otherwise identical in terms of game organization and support. Overall we observed more than 470,000 unique characters forming over 15000 guilds, scattered on three servers: Eitrigg (a PvE server), Cenarion Circle (a RP server), and Bleeding Hollow (a PvP server).

Social interaction may be an important influencing factor in guild-quitting events. First, we define a friendship network among guild members, where nodes are characters, and edges indicate co-occurrence within gaming zones — if two characters was observed in the same game location (zone in WoW), an edge is added

| Statistic on server Eitrigg | |
|---|---|
| Number of Characters | 51,224 |
| Number of Guilds | 2906 |
| Number of Edges | 2,447,577 |
| Average Collaboration Time (hrs.) | $1.73 \pm 1.09$ |
| % Characters changing Guild | 26.53 |

Table 1: Overall Network Statistics for the Eitrigg Server

between the corresponding nodes. The underlying assumption is that if characters co-occur in a gaming zone, it is highly likely that the characters are collaborating on a gaming activity. Two possible limitations are noted: (1) there are some gaming zones not necessarily associated with any gaming activity, for instance, characters are often left "AFK" (Away from keyboard) in the game's main cities before or at the end of a play session. In this case, the geographic proximity does not necessarily reflect any kind of joint activity. In our data logger, we remove such ambiguous zones from the co-occurrence criteria. (2) Characters may co-occur by chance. This is treated as noise in the social network graph. The basic assumption is that with large amount of accumulated gaming data, the ties between characters driven by real social interaction will dominate.

Secondly, we add a membership network to indicate the affiliation between characters and guilds. Nodes fall into two categories: (1) guild nodes, and (2) character nodes. If a character is observed appearing in a guild, an affiliation edge is added. The overall network is the super-imposition of the friendship and the affiliation networks. It is an undirected multi-graph i.e. it allows for multiple edges between any two nodes in the network.

The social network graph above has a temporal dimension: each edge has a weight indicating the duration of social interaction, and is tagged with a timestamp. We use a graph summarization approach as described in [124] to simplify representation into temporal snapshots. Given link weights $W_1, W_2 \ldots W_t$, the exponential kernel can be computed as follows,

$$W_t^S = \begin{cases} (1-\theta)W_{t-1}^S + \theta W_t & \text{if } t > t_0 \\ \theta W_t & \text{if } t = t_0 \end{cases} \tag{1}$$

where $t_0$ is defined as the initial time. The exponential kernel weighs the recent past highly and decays the weight exponentially as time passes, with the decaying rate defined by the parameter $\theta$. The summarized graph will be used for feature computing in the later sections.

Table 1 lists some statistics in the raw social network on Eitrigg. Guild quitting events are fairly common — around 26% of characters quit from a guild at least once in our observation period. Similar guild quitting statistics are observed on Cenarion Circle and Bleeding Hollow.

## 2.3 Are Quitting Events Correlated?

A common perception in WoW and other MMOGs is that social interactions between players influence a player's decisions about joining and quitting guilds. When a quitting event happens, the quitting player may pull friends out of the original guild. In this section, we examine the data formally to test this hypothesis. In WoW, the question to ask is, does data suggest that social interaction leads to correlated guild departures, or that on the contrary, departures are independent? This is the "sanity-check" question that should be addressed before diving into the social group analysis. If quitting events are independent, then social group analysis will be irrelevant and should not be the topic of our study.

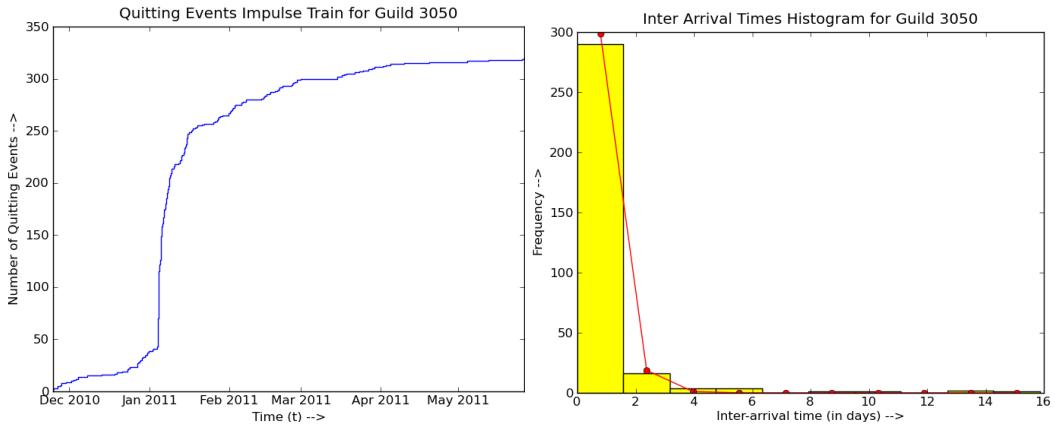



Figure 2: Quitting Events Impulse Train (left) and Distribution of Inter-Arrival Times (right) for a Guild on the Eitrigg server.

| Server | Number of Guilds | Number of Guilds with $> 30$ quits | Following Poisson Process | *NOT* Following Poisson Process |
|---|---|---|---|---|
| Eitrigg | 2906 | 181 (6.23%) | 23 (12.71%) | *158 (87.29%)* |
| Bleeding Hollow | 3425 | 309 (9.02%) | 28 (9.06%) | *281 (90.94%)* |
| Cenarion Circle | 2911 | 102 (3.50%) | 28 (27.45%) | *74 (72.55%)* |

Table 2: Guild Quitting Events Analysis

To test the correlated guild-quitting hypothesis, we model the quitting events as a Poisson process, a well-known probabilistic model for discrete events arrivals. Under the following assumptions, an arrival process is Poisson [129]:

1. The probability that one arrival occurs between $t$ and $t + T$ is proportional to $T$ (the proportion $\lambda$ is known as the arrival rate)

2. The number of arrivals in non-overlapping intervals are statistically independent;

3. The probability of two or more arrivals is negligible when $T$ is small.

We argue that assumptions 1 and 3 are intuitive and reasonable. We setup a statistical test for validating assumption 2.

It is known that, if the events follow a Poisson process, the inter-arrival time, defined as the time between two consecutive events, would be exponentially distributed with probability density function

$$f(T) = e^{-\lambda T}, \tag{2}$$

where $\lambda$ is the arrival rate that can be estimated from the observation data. This gives us a method to test the validity of Poisson models. To determine whether the probabilistic model (2) fits the observation data, we use the standard chi-square goodness-of-fit test [105]. To ensure statistical reliability, we limit our analysis to guilds with 30 or more departures. A guild is considered Poisson if the model fits with a confidence of 0.95 or higher. Figure 2 demonstrates the analysis for one such guild. Table 2 summarizes the result of goodness-of-fit test. A large fraction (87%) of guilds on the Eitrigg server (& other servers) do *NOT* follow a Poisson Process. This indicates that quitting events are not independent, but rather correlated. Factors that may have caused the correlation are explored in the later sections.

## 2.4 Potential Damage of a Quitting Event

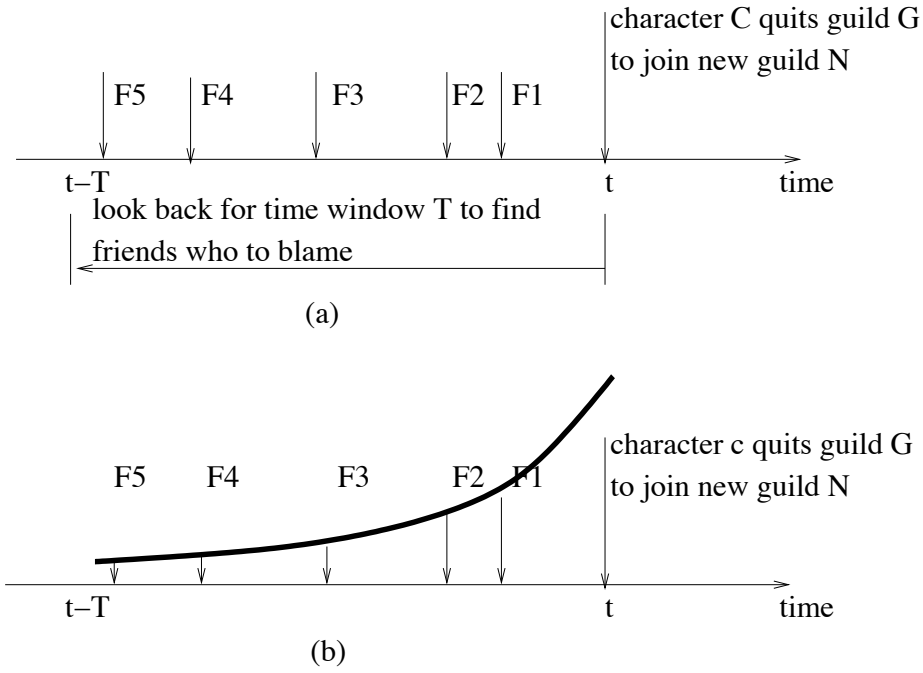



Figure 3: Damage Assessment

In WoW, the decision to quit a guild is likely influenced by social factors — unhappy players convince friends to join them in quitting, or a circle of friends reaching consensus to leave as a group. Modeling the details of this influence process is difficult without direct observation of player interactions. We adopt an abstract model in which a character's quitting decision is influenced by all preceding quitting events among his or her friends (as defined by the social graph) in proportion to the time elapsed. The immediate corollary is that a player who quits is responsible for, or shares the blame for every friend that subsequently quits the guild. We use the term *damage score* to refer to a character's aggregate share of the blame for subsequent quitting events. Essentially, damage score is an empirical notion of impact.

Figure 3 illustrates the computation of damage score. For a given quitting event (character $C$ quitting guild $G$ at time $t$ to join new guild $N$), we first look back in time to identify friends who may have caused this quitting event. In Figure 3a,

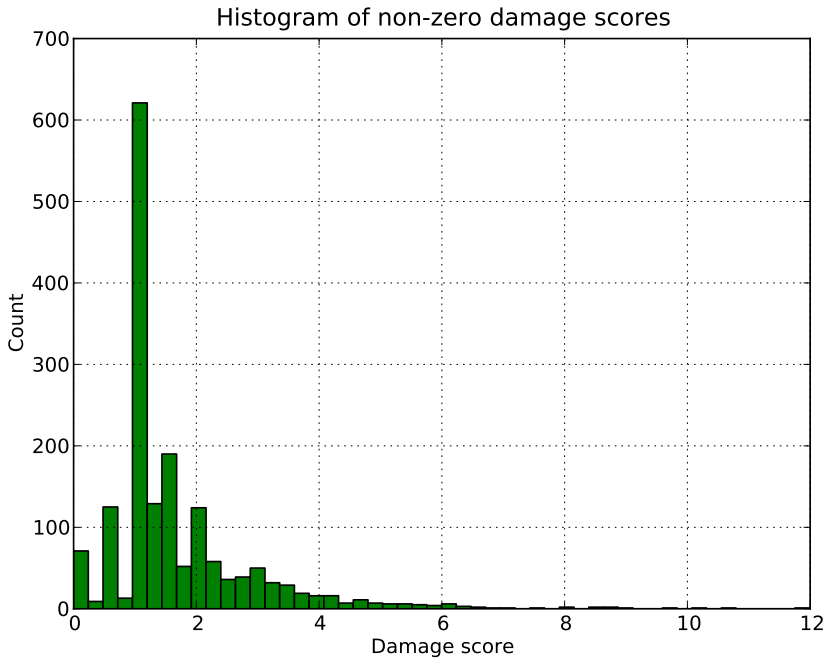



Figure 4: Histogram of Non-Zero Damage Scores

we note that a number of $C$'s friends ($F1, F2, F3, F4, F5$) all quit $G$ to join $N$ recently, hence $C$ may be under their influence. We then assign a "blame" score $b_{F,C}$ to the friends $F \in \mathcal{F} = \{F1, F2, F3, F4, F5\}$. Quitting events in the recent past receives high blame, while quitting events in the distant past is assumed to have little impact and receives little blame. In Figure 3b, $F1$ receives the highest blame, while $F5$ receives the lowest. Mathematically we use a normalized exponentially decay function:

$$b_{F,C} = \frac{e^{\alpha(|t_F - t|}}{\sum_{i \in \mathcal{F}} e^{\alpha(|t_i - t|}}, \tag{3}$$

where the parameter $\alpha$ controls how fast the blame function decays over time. The denominator normalizes the blame score so that the total blame across $\mathcal{F}$ sums up to 1.

Given the blame assignment, we now compute the damage of any given character $X$'s quitting as the sum of blame scores $X$ receives from all its followers

$C \in \mathcal{C}$, i.e.,

$$d_X = \sum_{C \in \mathcal{C}} b_{X,C} \tag{4}$$

For simplicity in implementation, $\mathcal{C}$ is defined over a time window $T$ after $X$'s quitting. Going further is unnecessary, as the quitting events are too far apart to assign or receive substantial blame. In our experiments, we use $\alpha = 1.0$ and $T = 15$ days.

In our observation data, the overall damage statistics is the following: among 23014 quitting events documented in the WoW Eitrigg server (PvE), only 1700 quitting events have a non-zero damage score. This indicates that most quitting events are isolated. Figure 4 shows the histogram of non-zero damage scores. Overall statistics are as follows: mean= 0.1232, standard deviation = 0.5581, min = 0, and max= 11.9911. Now the question is, from the observation data of a character's activity in WoW census, can we predict the potential damage if he/she leaves the guild?

## 2.5 Predicting Potential Damage

### 2.5.1 Features

For damage prediction, we take a supervised learning approach, where a feature set is generated and a mapping from the feature set onto the potential damage score is learned from a training set. Damage is inflicted by a character's quitting event (say, $X$ quitting guild $G$) to the guild. Several categories of features may be relevant: (1) individual player $X$'s profile, (2) the guild $G$'s profile, (3) game activity of $X$, in the WoW space as well in the guild $G$, and (4) social interactions and structural importance. The features are listed in Table 3.

Most feature names are self-explanatory. A few are explained below. For guild profile features (the second feature block in the table), two clustering coefficients are evaluated: the clustering coefficient of the guild, and the clustering coefficient of the entire friendship network. Clustering coefficient is a topological concept, measuring the degree to which nodes in a graph tends to be clustered together. Formally it is measured as the ratio between the number of closed triplets over the

| Feature Category | Feature Name | Correlation coefficient with damage score | Correlation coefficient with binary damage label |
|---|---|---|---|
| individual profile | character level | 0.1557 | 0.1591 |
| | number of guilds joined | -0.0423 | -0.0457 |
| guild profile | guild size | 0.0499 | 0.0558 |
| | clustering coefficient in guild | 0.0523 | 0.0669 |
| | overall clustering coefficient | 0.0369 | 0.0442 |
| game stats | playing time | **0.2651** | **0.3022** |
| | playing time within guild | **0.3581** | **0.4081** |
| | collaboration time | **0.2849** | **0.3274** |
| | collaboration time within guild | **0.3470** | **0.3991** |
| | collaboration coefficient | 0.1601 | 0.1747 |
| | collaboration coefficient in guild | 0.1393 | 0.1521 |
| | loyalty coefficient | 0.0770 | 0.0871 |
| social features | number of friends | 0.2383 | **0.2752** |
| | number of friends already having quit guild | **0.3918** | **0.4436** |
| | percentage of guild members played with before | 0.1789 | 0.1925 |
| | percentage of guild members played with excessively | 0.1781 | 0.1800 |
| | weighted degree | **0.2673** | **0.3189** |
| | weighted degree in guild | **0.2618** | **0.3091** |

Table 3: Correlation Coefficient between Features and Damage Score. Correlation Coefficients with absolute value exceeding $0.25$ are marked in bold-face fonts.

number of connected triples of vertices. It takes value 0 in a star topology and value 1 if the graph is fully connected. Through these features, we would like to investigate whether structure balance has an impact on potential damages of quitting events.

In the game statistics features (the third feature block in the table), in addition to playing time, our method logs collaboration time, the time spent by character $X$ in playing with other characters (in the entire WoW space and within the guild). In addition, we have a few features indicating the playing style of a given character. Collaboration coefficient is defined as the ratio between the collaboration time and the overall playing time. A high collaboration coefficient indicates a social playing style while a low value implies a "lonely wolf" player. Loyalty coefficient is defined as the ratio of $X$'s time in guild $G$ and $X$'s total playing time. This measure indicates how loyal $X$ has been to the guild $G$ (prior to quitting).

Social features measures the relative importance of the given character $X$ in the game space and the guild. Some friends have a lot of interactions. "Played with excessively" here is defined as those with collaboration time exceeding 2 standard deviations above the mean collaboration time among all pairs of characters. These are considered close friends. Weighted degree is a measure of centrality. It is computed from the temporally-summarized graph. We would expect that more central $X$ is, more damaging his/her departure would be.

### 2.5.2 Feature Importance

Table 3 also contains the correlation coefficient between each feature and the damage score (the third column). The correlation coefficient can be considered as a rough measure of the feature's importance in predicting damage. If the damage is independent from a feature (hence the feature is useless to prediction), the correlation coefficient will be 0. Sign of the correlation indicates when a feature is positively correlated (big feature value implies big damage) or negatively correlated (otherwise). Significant correlation coefficients (with absolute value exceeding 0.25) are marked in bold-face fonts.

Among individual player $X$'s profile features, character level appears important. High level players are more likely to cause significant damage. This is not

surprising. The number of guilds that $X$ has joined prior to the quitting event may be related to $X$'s willingness to join new guilds, but it does not appear to be indicative of how much damage $X$'s departure may cause to the original guild.

Guild profile features do not seem to be very strong features. Both size and topology (measured by the clustering coefficients) seems to be weak features.

Game activity features are of high importance. Total playing time, in the entire WoW space and within the guild, are both strongly positively correlated with damage. This agrees with the intuition — the more $X$ plays, the more important he/she becomes in the guild, and hence $X$ quitting the guild is likely to cause more damage. Collaboration time in the WoW space and within the guild are also strongly positively correlated with damage. The relative ratios, collaboration coefficient and loyalty coefficient, are not as strong. This can also be justified — being loyal to the guild simply means it is hard to quit, however it does not necessarily imply low damage.

Social features are very important. The centrality measures (weighted degree features, number of friends, etc) are strongly correlated with damage. This agrees with our intuition that the departure of a central node can be very damaging because it can cause a snowballing efficient. Furthermore, the number of friends of $X$ who have already quit the guild is very important. This may be justified from a social psychology perspective – if many of $X$'s friends have already quit the guild, then there is a tendency that more friends will join in the quitting streak. This boosts up the potential, and when $X$ quits, the snowballing effect can accelerate.

### 2.5.3 Prediction of Damage Significance

We first formulate damage prediction as a classification problem — is $X$'s quitting going to cause substantial damage to its original guild $G$? The class label in this case is binary: substantial if the potential damage score exceeds a pre-determined threshold (with value 1 in our experiment) and non-substantial otherwise. The last column of Table 3 lists the correlation coefficient between the features and the class label. Qualitatively it does not differ much from the previous column. Game activity and social interaction remain the strongest feature categories.

| Server | Accuracy | Precision | Recall | F-measure |
|---|---|---|---|---|
| Eitrigg | 82.50% | 0.825 | 0.825 | 0.825 |
| Cenarion Circle | 81.23% | 0.812 | 0.823 | 0.813 |
| Bleeding Hollow | 79.9% | 0.799 | 0.799 | 0.799 |

Table 4: Damage Classification. Results are reported on 10-fold cross validation over the training set.

One problem to note is that the observation data set contains an overwhelmingly large non-substantial damage class (93% of all samples) and a small substantial damage class. Out of the 23014 quitting instances, only around 1700 has a non-zero damage value. If we randomly select training samples, the predictive model will overfit in the small damage value interval, and may fail to predict high damage quitting events. To avoid overfitting, our approach takes a balanced sampling approach to generate a training set containing roughly an equal amount of samples from each class.

To predict the damage class label, we have experimented with a number of classification methods. Tree-based classification methods work well. Table 4 reports classification results using a Weka implementation [58] of the random forest method, which builds a library of decision trees from random feature subsets and predicts class labels by voting or averaging. The overall classification accuracy is around 80%for all WoW servers. This is significantly better than random guessing (with around 50% accuracy). This result indicates that our feature set is predictive of future damage, and the classifier can be used to predict damage labels reliably.

### 2.5.4 Regression For Damage Prediction

Damage prediction can also be formulated as a regression problem, i.e., constructing a mapping from the feature set to the continuous damage value. Regression performance is measured using common metrics such as mean absolute error (MAE) and mean squared error (MSE). The correlation coefficient between the predicted value and the true target value can also be considered as a metric of accuracy, with high value indicating good prediction performance.

Table 5 lists regression results using a Weka implementation [58] of Bagging

| Server | MAE | MSE | Correlation Coefficient |
|---|---|---|---|
| Eitrigg | 0.6836 | 1.0830 | 0.7046 |
| Cenarion Circle | 0.6032 | 0.9140 | 0.6729 |
| Bleeding Hollow | 0.6853 | 1.0798 | 0.6495 |

Table 5: Regression from feature set to damage (evaluated on the training set via 10-fold cross validation)

method [61], which builds a library of regression trees and averages their prediction results. For Eitrigg, the MAE is 0.6836 (in a dynamic range of 0 to 11.99). and the MSE is about 1 (in a dynamic range of 0 to 143.7). This indicates that regression accuracy is quite reasonable. Similar regression accuracy has been observed on Cenarion Circle and Bleeding Hollow as well.

We conclude that the feature set combining individual profile, guild profile, game activity statistics, and social interactions can support damage prediction, and that our approach can predict damages associated with quitting events with reasonable prediction and classification accuracy.

## 2.6 Predicting Guild Quitting Events

In this section, we examine the feasibility of predicting if and when a character might quit his or her guild. Prediction of guild quitting behavior has both practical and theoretical value. From a theoretical point of view, identification of factors that predict guild quitting events gives us insight into factors affecting guild stability. From a practical point of view, we can build a useful detector for identifying characters who might potentially quit and abscond with guild resources or otherwise damage the effectiveness of the guild. We might then be able to intervene before adverse outcomes are experienced.

The prediction problem is formulated as the following: given the game trace up to current time, predict whether a character will quit from the guild within a specified future interval. In WoW, there are two notions of time: (1) calendar time, measured in hours and days, and (2) game time, measured as the count of game sessions and events. Here we unify these two notions by grouping game events

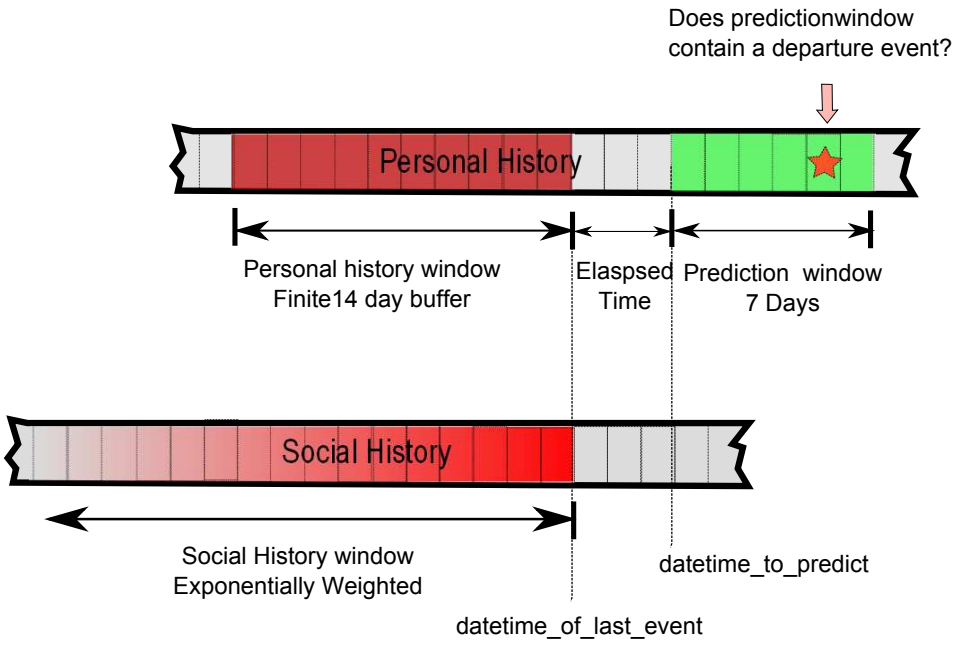



Figure 5: Personal and Social History Windows and the Prediction Window

within a guild by day and call it a "day record". Usually there is one record per day, but if the player switches guilds, there may be more than one day record for a specific calendar date, and if the player does not play at all, there will be no day record. The day record notion is flexible and can accommodate variation in playing style, ranging from players with only occasional game activities to more serious players. Based on this notion, we define a prediction window in terms of day records (7 in our experiment). If this window of future day records includes observations of the character in another guild, we set the class label *target_guild_-has_changed* to true to indicate a quitting event has occurred. This class label is our prediction target.

### 2.6.1 Features

Prediction of quitting event is inherently dynamic. Game activities unfolds as time advances, and prediction should be updated accordingly. To accommodate the dynamic nature, we keep a running window of features. A personal history window of 14 day records is used to generate features summarizing the character's playing

| Feature | Correlation coefficient |
|---|---|
| *guild_count* | 0.183 |
| *time_since_last_event* | 0.064 |
| *event_count* | -0.056 |
| *level_begin* | -0.051 |
| *level_end* | -0.050 |
| *level_change* | 0.041 |
| *avg_event_duration* | 0.014 |
| *window_duration* | 0.001 |
| number of guild memberships | 0.170 |
| clustering coefficient within guild | -0.151 |
| playing time within guild | -0.122 |
| collaboration time within guild | -0.107 |
| weighted degree within guild | -0.105 |
| overall clustering coefficient | -0.096 |
| overall weigted degree | -0.089 |
| playing time | -0.089 |
| collaboration time | -0.089 |
| percentage of members played with excessively | -0.082 |
| percentage of members played with before | -0.078 |
| number of friends | -0.062 |
| number of guild members | -0.025 |
| number of friends already having quit guild | 0.007 |

Table 6: Correlation between features and quitting events. Personal history features are shown italicized, and social history features are in regular fonts.

history, as shown in Figure 5. The features are listed in Table 6 in italicized fonts in the first feature block. Basically, these features are designed to measure game engagement and achievement within the history window. The guild count represents the character's past tendency to quit his or her guild. The character level and level change attributes are intended to capture the character's sense of progress. The underlying theory being that a character that is making progress will be content with their current guild. The number of game events in the window and duration of the window are designed to help the classifier address special cases that occur for many

characters when we are predicting at the beginning of their histories. Average event duration is designed to capture how dense the character's play is. Time since last event measures the gap since the last historical game session.

From a social psychology point of view, a person's decision to leave a guild may be primarily due to the unsatisfaction towards the guild. Hence prediction should not only involve individual character's personal history, but also features regarding the guild, especially social relationship with guild members. Table 6 contains these features as well (second feature block, in regular fonts). Details regarding these features are explained in the last section and hence are omitted here. The social features are computed from the temporally summarized graph using an exponentially decaying summarization kernel. This is equivalent to a "fading" social history, as illustrated in Figure 5.

Table 6 sorts the two feature categories according to relative feature importance, measured as the absolute value of correlation coefficient. *Guild_count* (and equivalently the number of guild membership) seems to be the strongest feature. This is intuitive — pass switching events is indicative of a character's lack of loyalty, and implies an easy tendency to depart from his/her current guild as well. The feature *time_since_last_event* is positively correlated with quitting, i.e., the longer a player stays activity free, the more likely he/she will quit the guild. Among the social history features, clustering coefficient (measuring structure balance), playing time and collaboration time (measuring engagement within the guild), and weighted degree (measuring importance) are all negatively correlated with guild departure. This is also intuitive. A person heavily engaged and playing a central role in a well-balanced guild is less likely to leave the guild.

We note the related work in [20] which primarily relies on individual features to detect churn events, and the influence model [9] which only considers social context. The novelty of our approach here is that we have identified factors from both categories and their relative importance. As Table 6 shows, important features comes from both categories and should be unified for prediction. We have tried Weka's linear forward greedy feature selection to find an informative feature subset. The subset returned contains two personal features (guild_count, time_since_last_event) and five social features (playingTimeWithinGuild, collaborationTimeWithinGuild, numberOfGuildMembers, numberOfGuildMembership,

clusteringCoefficientWithinGuild).

### 2.6.2 Method and Results

In this prediction problem, the prediction events for a server are not completely independent. Successive predictions for a given character will be correlated. This can lead to overfitting where the system learns to identify specific characters who quit instead of generic features of characters which are useful for predicting quitting events. To guard against this, we developed distinct training and test sets. History and prediction windows in the training set are generated from a different set of characters than the history and prediction windows in the test set.

| Server | Overall | Non Quitting Event | | | Quitting Event | | |
|---|---|---|---|---|---|---|---|
| | Accuracy | Precision | Recall | F Score | Precision | Recall | F Score |
| Eitrigg | 82.7432 | 0.878 | 0.926 | 0.901 | 0.389 | 0.268 | 0.317 |
| Cenarion Circle | 89.0973 | 0.917 | 0.967 | 0.941 | 0.342 | 0.164 | 0.222 |
| Bleeding Hollow | 79.8396 | 0.855 | 0.91 | 0.881 | 0.396 | 0.276 | 0.325 |

Table 7: Results on balanced training set tested on disjoint character IDs

Guild quitting prediction is further complicated by the unbalanced class problem. In most of the observation snapshots, characters continue to play in their current guild, and quitting events are rare by comparison. To avoid overfitting the classifier to non-quitting events, we use a random sampling method to balance the training set so that it contains approximately equal proportions of non quitting and quitting events. Classification results were similar for a number of prediction models. Table 7 reports the classification performance for a random forest with 10 trees and unlimited depth and feature counts. Guild quitting prediction classifiers are built separately for 3 WoW servers: Eitrigg, Cenarion Circle, and Bleeding Hollow. Classification performance does not vary much on servers, indicating that our approach is generally applicable in the WoW space.

We conclude that it is possible to predict with modest precision and recall and that past loyalty, guild stability, and social engagement are key predictive factors for quitting.

## 2.7 Conclusion

Our analysis has shown that destructive group dynamics models can be constructed from WoW in-game census data. We have build predictors to predict (1) the potential impact of an imminent guild departure and (2) if and when the depart will happen in a prediction window. The predictors have reasonably accuracy (best predictors in the 80-90% accuracy range). It is important to choose features from multiple perspectives, such as features regarding the individual, guild features, game activity statistics, and social interaction and topological features. Combining diverse features is essential to the predictor performance.

Our future work will likely include extending group dynamics modeling and prediction to constructive dynamics to understand how a player joins a guild and how a guild grows. More importantly, we would also like to extend the group dynamics analysis to other social groups, for instance, real-world social interactions, and online social networks.

# Chapter 3

# Predicting Group Stability in Online Social Networks

## 3.1 Introduction

Understanding community structures has always been an interesting topic in social sciences. In many social network datasets, a social graph is presented in which nodes represent individuals and edges represent social ties. It is a common experience to observe community structure in such a graph, in the sense that a subset of vertices are well connected within them and less connected to the rest of the graph. For example, communities in a social network often represent social groupings, say by interest or background. Communities in a publication network may represent people who work on similar research problems. Communities of the web graph may suggest pages on related topics. As the complexity of online activity increases, formal group structures have come to play an increasingly important role in the experience and effectiveness of an individual's online life.

Online platforms have provided unprecedented opportunities to study large-scale behavior and dynamics of communities. A lot of studies have focussed on how to define and detect social communities in the network structure and how the groups evolve over time. For the later, the main thrust of such research has been to model the evolution of groups, from the standpoint of growth [12, 99, 155]. A community in these studies always grows. What to be examined is the rate of growth and

when the community stops growing. There are two main reasons for this. First, in many online social network settings there is no restriction on the number of groups an individual belongs to. Also, in most cases individuals do *not* quit groups even though they may not be active participants in those particular groups. This often results in groups having a monotonically increasing membership curve throughout their lifetimes. In addition, a practical and commercial motivation for such studies has been to increase the 'stickiness' of an online community, i.e., the capability for it to attract new members. Therefore, a common model in modeling group growth is to consider it as a diffusion process. That is, the social ties that cross group boundaries may influence people not yet in the group to join the group. This observation has been one of the main philosophies in modeling and predicting the growth of online groups. Studies have been performed in examining how diffusion happens and what is the main factor in determining the speed of diffusion. It has been shown using Facebook data [1] that what attracts a new user depends not only on the number of friends on Facebook, but also on the diversity of these friends, as well as the network connectivity structure among them [132].

In this chapter, we take a different perspective and study the complementary problem of group stability, i.e., why some groups fall apart and disappear while others thrive. The effectiveness of groups can be undermined when group members depart, taking with them, experience, resources and possibly other group members. The ability to predict the stability of groups is highly desirable, as it offers insights on factors that affect online group effectiveness. It also provides practical guidance to tasks such as risk management and customer retention.

In some online settings an individual can belong to *only* one group at any given point in time. In such settings the group serves as the main engagement platform for the individual. An individual who is not satisfied with his/her group will quit the group and join another one. The reasons for dissatisfaction can be plenty. In such cases the percentage increase/decrease in the number of group members over previous time periods is a good measure in determining whether a group is stable or shrinking. In the settings when users can join multiple groups and probably never quit these groups, the group size always grows. But the growth in group size does

[1] `https://www.facebook.com`

not necessarily capture the accurate picture. A group, though accumulating members over time, may still be a shrinking group, if most members do not participate in the group's activities. In addition, most previous studies treat all group members as equals when performing group evolution analysis. We know that groups are often led by a smaller set of leaders who have considerable influence over other group members. In our analysis we take both issues into consideration. For settings that allow multiple group memberships and do not have group quitting events we devise a membership score that will reflect participation level of individual members and prolificness/ranking of individual members.

We perform our analysis on two different types of social networks, a massive multiplayer online role-playing game (World of Warcraft [WoW]) and a large co-authorship network (DBLP). In the first dataset we tackle the scenario of an individual belonging to at most one group (a guild, in WoW terminology) at any given point in time and the second tackles the more general scenario of an individual belonging to multiple groups at any given point in time. Moreover for the second scenario, we also devise a membership score that we believe is more reflective of the stability/growth of a group (as compared to the number of members in the group). This membership score can be easily generalized for a host of social networks. Though the membership score was devised to encapsulate the growth (or lack of) of a group it can also be used to compare groups, as we will demonstrate later. We have built classifiers based on a diverse set of features to predict whether a group will have significant reduction or will remain stable over a period of time.

In our findings regarding the two datasets, we have the following interesting observations:

- We find that the level of diversity has a strong correlation to the stability of the group. In order to keep a group alive, members of the community should vary in terms of expertise, seniority, responsibilities, etc. We also find that the level of activities has a strong predictive power of the group stability. Even when the size of the community stays the same (i.e., not attracting new members), as long as there is a lot of activities within the group the community survives.

- We find that in the case of WoW dataset the age of a community has a strong correlation with the stability of the group — if a guild can sustain itself for

a long period of time, it is very likely that the guild does have the essential components necessary for a stable community. On the hand in the DBLP dataset the length of existence does *not* show any correlation with group stability. Whether a conference is old or new does not seem to play a significant role in determining whether it remains stable or shrinks. This observation can be attributed to fact that in WoW there is a lot of churn, whereas in DBLP (or other related social networks) we do not see as much churn.

- For DBLP dataset we observe that the 'average prolificness' feature is important. The correlation shows that groups with more prolific members are more likely to remain stable and groups with more dedicated authors (i.e. authors who continually contribute) are more likely to remain stable. Thus such members play an important role in maintaining the stability of the group.

The chapter is organized as follows. We first briefly review literatures on detection of communities and studies of community evolution. We then provide an overview of the datasets, followed by definition of measures used to label groups as stable or shrinking. We then move on to define a range of features that we compute for both datasets. We will also analyze the best set of features that are useful for our prediction task. The later sections present predictive models to predict group stability. We conclude with a discussion of the important factors for predictions and an outline of future work.

## 3.2 Related Work

Various methods have been used to detect communities. See the survey paper [116] for a thorough review. Earlier approaches define some measurement of importance of each edges and then define communities by either incrementally adding edges in the order of decreasing importance [140]; or removing edges in the order of increasing importance [53]. This leads to a hierarchical partitioning of the nodes, called a *dendrogram*. Classical clustering techniques are also used here, including $k$-means clustering, multi-dimensional scaling, principal component analysis, etc. Same for methods that identify clique-like components, or find min-cuts in graphs.

In our datasets, community structures are formally defined and explicitly given, hence there is no need to detect them.

When time-stamped data is available it is natural to ask how the communities or groups evolve over time. In the literature the community evolution has been modeled as a diffusion process – ties spanning group boundaries can possibly influence individuals to join the community. Granovetter [56] pointed out that diffusion often benefits from 'weak ties' and indicated that the graph structure may be a critical factor in deciding whether and how fast a community grows. Recent studies, such as by Centola and Macy [30] and from Facebook datasets [132], revealed that one may require multiple contacts within the group to join the community and the diversity of these contacts actually matters. A couple of the analysis using real world datasets show conflicting and intriguing observations that high clustering property inside a community may at the same time both attracts new members and prevents overall growth. A very recent paper by Kairam *et al.* [75] pointed out that new members may join through diffusion (as in the case of being influenced by some friends), or may join the community without having any social ties inside the community, classified as non-diffusion growth. They further point out that in diffusion-based growth, the clustering does help. But groups that only grow through diffusion may not reach large size. Thus non-diffusion growth is important to create large communities.

Most of existing work on community evolution focused on the initial stage of community evolution, when growth is in the dominant form. Our work, on the other hand, mainly looks at the final stage of community evolution, i.e., how a community dies or falls apart. The closest work to ours is the research on group formation in large social networks [12]. They build classifiers based on a range of network-based features to firstly, predict whether an individual will join a community and secondly, to predict the growth of a community. They achieve reasonably good $(70 - 75\%)$ accuracy for both prediction tasks. The point to note is that members never quit communities in their model. Thus, for the second prediction task they are predicting from the standpoint of growth. Our work is complementary to their work. In our previous work in chapter 2, we have built models to predict if and when an individual is going to quit his/her group, and whether this quitting event will inflict substantial damage on the group. We quantify damage as influencing

many of your friends to also quit the group after you do so, thereby leading to a large loss in group membership numbers. In chapter 2, we analyzed quitting from an individual perspective, while this chapter addresses the quitting behavior from a group perspective.

## 3.3 Analysis on World of Warcraft

### 3.3.1 Dataset

| Statistic | Eitrigg | Bleeding Hollow | Cenarion Circle |
|---|---|---|---|
| Number of Characters | 51,224 | 72,108 | 47,499 |
| Number of Guilds | 2906 | 3425 | 2911 |
| Number of Friendship Edges | 577,250 | 937,989 | 673,502 |
| Number of Membership Edges | 1,870,327 | 2,775,401 | 2,154,287 |
| Avg. Collaboration Time (hrs.) | $1.73 \pm 1.09$ | $1.70 \pm 1.10$ | $1.79 \pm 1.08$ |
| % Characters changing Guild | 26.53 | 32.28 | 20.69 |

Table 8: Overall Network Statistics for World of Warcraft

We use the World of Warcraft (WoW) dataset that formed the basis of our study in the previous chapter. Section 2.2 provides a detailed overview of the data and how we use the data to construct the gaming social network. Overall we observe more than 470,000 unique characters forming over 15000 guilds, scattered across three servers: Eitrigg (a PvE server), Cenarion Circle (a RP server), and Bleeding Hollow (a PvP server). Table 8 lists some statistics in the raw social networks for the three servers. Guild quitting events are fairly common — around $20 - 32\%$ of characters quit from a guild at least once in our observation period. In constructing our social network using the co-occurence heuristic, we eliminate characters that do not join any guilds or collaborate with any other characters (i.e. characters that do not have any social component). Such characters are generally played by new players at initial levels of the game. These characters are uninteresting from the point of view of the problem we seek to tackle and thus can be ignored from our

analysis. Similarly we ignore degenerate guilds (i.e. guilds which are observed to have no members over our entire 6 month time-period) from our analysis.

### 3.3.2 Gauging Group Stability

It is often of general interest to understand the stability issues of social groups, for instance, the stability of a company, an informal organization, or a user group. In WoW and other MMORGs, as mentioned earlier, guilds have high turn-out rates. In our observation data of over 6 months, some guilds live throughout (188 days), but many other do not survive very long. The average guild lifespan is 82.57 days, with a large standard deviation of 71.25 days. This begs the question, "Why are some guilds more stable than others? In other words, what constitutes a stable guild?"

Guild stability may be related to a variety of factors, some of which have been identified from social psychology studies. In this section, we take a data driven approach — "Can we identify stability or instability patterns from the data?"

It turns out that since a character can only belong to one guild at any given point in time, computing the number of guild members at regular intervals should give us a good idea of how the guild evolves. Futhermore computing the percentage of increase/decrease in the number of members over the previous interval would then give us an accurate idea of whether, (a) the guild is stable (i.e. there is a minimal percentage decrease or a percentage increase in the number of members), or (b) the guild is shrinking (i.e. there is a substantial percentage decrease in the number of members).

To put things more formally, given that a guild $G$ has $m_1$ members at time snapshot $t_1$ and $m_2$ members at time snapshot $t_2$ ($t_1 < t_2$) the percentage change in membership is defined as $\delta = \frac{m_2}{m_1} - 1$. We could then label a guild as being in the stable or shrinking phase at time $t_2$ as follows,

$$label = \begin{cases} stable, & \text{if } \delta > -0.15 \\ shrinking, & \text{if } \delta \leq -0.15 \end{cases} \tag{5}$$

Thus we label a guild as shrinking if it loses $15\%$ or more of its members as compared to the previous interval, otherwise the guild is labeled as being stable. In our experiments, we have experimented with several values of $\delta$. Results were comparable when $\delta$ ranged between $[0.10, 0.20]$. For $0 < \delta < 0.10$, accuracy was reduced

due to addition of noisy/fringe samples to the "shrinking" set. For $\delta > 0.20$, we again observe a drop in accuracy due to vastly fewer "shrinking" samples. Thus, in our experiments, we use a threshold of $\delta = 0.15$ to label whether a guild is stable or shrinking at any given point in time.

### 3.3.3 Guild-Level Features

In order to be able to model group stability dynamics, we consider a range of features that span different categories. Almost all of the features are efficient (with linear running time) to compute thereby allowing us to compute them at regular intervals and making our approach scalable to large networks.

Several types of guild-level features may be important in modeling guild stability. We loosely categorize them into three categories: (1) guild composition, (2) game activities aggregated over the guild population, and (3) the structure of social network graph.

Guild composition features reflect diversity or homogeneity of guild members. In WoW, guilds need to have a certain span in skills and roles. For instance, a healthy blend of experts and novices may be important to a guild's long-term survival. Novice players can mature in the game and take over if an expert leaves the guild. In addition, WoW activities are designed to encourage collaboration across roles. Characters are categorized into 10 classes (warriors, paladins, hunters, priests, death knights, etc) with different capabilities (DPSs to cause damage, tanks to contain damage, and healers to heal damage). Coordination among the classes and capabilities is essential to the guild's success. There is a "sweet spot" of diversity, where, a certain degree is desirable, while excessive diversity may be a sign of lack of management and may imply poor guild performance. To investigate the effect on guild stability, we compute the following guild composition features:

**Number of guild members:** The size of the guild at time $t$.

**Length of existence (in days):** This feature calculates the number of days since the guild came into existence.

**Average level of guild members:** This measures the average level of characters who are members of this guild at time $t$.

**Standard deviation of character levels of members:** This feature measures how consistent good/bad characters are across the guild. A smaller standard deviation indicates a bunch of members with equal skill sets and a higher standard deviation indicates a bunch of members with varying skill sets.

**Percentage of character classes present:** A character can belong to any one of around 10 character classes. This metric measures whether all character classes are represented amongst its members.

**Entropy of character class distribution:** This feature calculates the entropy of the class distribution for the given guild.

**Entropy of character category distribution:** There are 3 categories of characters in the game, DPS, Healers and Tanks. Each character class can perform one or more of these category roles. This feature calculates the entropy for category distribution within a guild.

Aggregated game activity features measures the overall game engagement across guild members. We contrast the game activities within the guild and prior to joining the guild. The former is indicative of the devotedness to the guild, while the latter measures the overall engagement in the entire WoW game.

**Average playing time within guild & prior to joining guild:** The features calculate average playing time of all guild members in the current guild and prior to joining the guild respectively.

**Average collaboration time within guild & prior to joining the guild:** The features calculate the average collaboration time of all guild members in the current guild and prior to joining the guild respectively.

**Average collaboration coefficient within guild & prior to joining guild:** The collaboration coefficient for a guild member is defined as the ratio of his/her collaboration time to playing time. These two features compute the average collaboration coefficient across all guild members by considering collaborations within the guild and prior to joining the guild respectively.

We suspect that guild topological structure may have implications on guild stability. Guilds exhibit remarkable diversity in topology, some guilds have a hierarchical structure, where some nodes (typically guild leaders) are of central importance, while other guilds are formed of closely knitted friendship circles, where nodes are more evenly connected. One may speculate that a star topology may be less stable since the removal of the center node may cause the whole graph to fall apart. In the topological features, we measure the average clustering coefficient, which is a metric of degree to which nodes in a graph tends to cluster together. Based on the concept, we compute the following topological features:

**Average clustering coefficient of guild members:** We measure the clustering coefficient at each guild member node and then calculate the average of this clustering coefficient across all guild members. The clustering coefficient at each node is also known as the local clustering coefficient [141] and quantifies how close its neighbors are to being a clique.

**Average clustering coefficient of guild members within guild:** The clustering coefficient calculated in this case only takes into consideration the graph induced by all members of the guild.

**Entropy of degree distribution:** This feature is a good measure of diversity in node connectivity.

#### 3.3.3.1 Feature Importance & Correlation

The observation data is organized into temporal snapshots, sampled every 4-day interval. Overall there are about 63000 guild-snapshots. Guild features (composition, game activity, and structural) are computed for each guild-snapshots. Furthermore, we label the data samples as shrinking or stable guilds. If a guild will lose more than 15% of its membership in 4 weeks, the guild at the current time will be labeled as "shrinking", otherwise the guild is labeled as "stable". This simplifies the guild stability problem into binary classification. Thus we are trying to predict 4-weeks into the future as to whether a guild will remain stable or will shrivel.

Random sampling is used for drawing training samples. There are more

| Category | Feature | Correlation Coefficient |
|---|---|---|
| Composition | number of guild members | **-0.1213** |
| | length of existence | **-0.1501** |
| | average level of members | 0.0037 |
| | standard deviation of member levels | -0.0649 |
| | percentage of character classes present | **-0.1153** |
| | entropy of character class distribution | -0.0707 |
| | entropy of character category distribution | 0.0087 |
| Game stats | average playing time in guild | **-0.1038** |
| | average playing time prior to guild | 0.0053 |
| | average collaboration time in guild | **-0.1025** |
| | average collaboration time prior to guild | -0.0019 |
| | average collaboration coefficient in guild | **-0.1026** |
| | average collaboration coefficient prior to guild | 0.0067 |
| Structural | average clustering coefficient | **-0.1021** |
| | average clustering coefficient in guild | **-0.1103** |
| | entropy of degree distribution | **-0.1288** |

Table 9: Correlation coefficient between class labels and feature values for Eitrigg Server. Correlation coefficients with absolute value exceeding 0.10 are marked in bold-face fonts.

shrinking guilds in the data than stable guilds, hence an uncontrolled random sampling may cause the classifier to overfit shrinking guilds. We control sampling to produced balanced classes, 2000 samples from each class. We would like to understand which features are important in modeling guild stability. Table 9 reports the correlation coefficient between each feature and the class labels (1 for shrinking, 0 for stable) for Eitrigg. Results for Bleeding Hollow & Cenarion Circle qualitatively agree with these results and hence we omit them for the sake of brevity.

**Among the guild composition features:** Large guilds tend to be more stable. Guilds which have survived for longer periods tend to continue to survive. These

agree with empirical observations and intuition. Average member level does not seem to matter much, however, diversity seems to play an important role. For instance, standard deviation of member levels are negatively correlated with guild shrinkage, indicating that diversity seems to help guild stability. Likewise, diversity in character classes is important. Guilds with more number of character classes present survive better.

**Among the game activity features:** In-guild activity matters a lot. The more guild members collaborate and play, the more stable the guild is. The activity of guild members prior to joining the guild does not seem to matter at all.

**Among the structural features:** All features seems to be very strong features. Balance and diversity in topology helps to improve overall guild stability.

| Rank | Feature | Category |
|---|---|---|
| 1 | number of members | Composition |
| 2 | entropy of class distribution | Composition |
| 3 | percentage of classes present | Composition |
| 4 | average collaboration time within guild | Game Activity |
| 5 | length of existence | Composition |
| 6 | average playing time within guild | Game Activity |
| 7 | entropy of degree distribution | Structural |
| 8 | standard deviation of member levels | Composition |
| 9 | average clustering coefficient within guild | Structural |
| 10 | average clustering coefficient | Structural |

Table 10: Top ten features for Eitrigg, ranked in descending order of information gain.

Another common method for assessing the relative importance of features is the mutual information between a feature and the class label. This indicates how informative a feature is. We use Weka [58], a machine learning toolbox. It provides information gain computation and rank the features. In descending order of information gain, the top ten features are listed in Table 10. Compared to correlation

coefficient analysis, the information gain ranking is more precise. It does not rely on single-mode distribution, which is an inherent limitation of correlation coefficient. However, the information gain does not reveal the insight that positive/negative correlation reveals. Qualitatively, Tables 9 and 10 are in rough agreement.

### 3.3.4 Predicting Guild Stability

Given the feature set and the class labels (stable or shrinking), we want to predict whether a group or community is likely to remain stable or will start shrinking over a period of time. We experiment with a range of supervised learning methods to achieve this.

With the feature set, we are able to predict guild stability with good accuracy. For instance, using guild size feature alone (number of members) and simple classification such as Naive Bayes, we can predict shrinking or stable labels with about 59% accuracy. Using two features for prediction, the number of members and length of existence, Naive Bayes produces a prediction accuracy of roughly 62%.

| Method | Classification Accuracy | Precision | Recall | F-measure |
|---|---|---|---|---|
| ZeroR baseline | 50% | 0.25 | 0.5 | 0.333 |
| Naive Bayes | 63.74% | 0.682 | 0.637 | 0.614 |
| Decision Stump | 62.10% | 0.651 | 0.621 | 0.601 |
| J48 decision tree | 78.86% | 0.790 | 0.789 | 0.789 |
| Bagging | 81.98% | 0.822 | 0.821 | 0.820 |
| Random Forest | 84.78% | 0.848 | 0.848 | 0.848 |

Table 11: Guild Stability Prediction Results

Table 11 summarizes the result of guild stability prediction using a variety of classification methods. The testing set is balanced, where an equal amount of testing samples from each class are randomly drawn from the observation data. The results are reported after a 10-fold cross validation process.

Classification methods include the following, with the first three methods serving as benchmarks.

**ZeroR baseline:** It is a degenerated classifier, always predicting a shrinking guild regardless of the features.

**Naive Bayes:** It assumes that all features are independent given the class label, and constructs a probabilistic model for each feature separately. The classifier computes likelihood from all features and chooses the maximum likelihood class label as the classification result.

**Decision stump:** It is an one-level decision tree making a prediction with just a single input feature. In our training data, the single feature is average collaboration time within guild. If it is more than 5.002 hours, the guild is predicted to be stable. Despite its simplicity, the prediction accuracy is decent, in the $60 - 70\%$ range.

**J48 tree [118]:** It progressively grows a decision tree by identifying the attribute that discriminates the training set most clearly according to an information gain criterion. The tree branch terminates if the training samples at the leave are homogeneous. The prediction accuracy of J48 tree is close to $79\%$.

**Bagging [24]:** Bagging is an ensemble method which improves the classification accuracy through sampling and model averaging. We get an accuracy of close to $82\%$ using Bagging.

**Random Forest [25]:** Similar to bagging, random forest is also an ensemble method. It builds a library of decision trees from a set of random samples. Each decision tree is grown by randomly choosing the variables to split data upon. The classifier predicts class label by average voting from the decision trees. This method works well when there is sufficient training data. The accuracy is around $85\%$.

One hypothesis regarding guild stability is the continuity — if a guild has been shrinking recently, it is anticipated to continue the loosing streak. This hypothesis has been raised in the literature of social network analysis. To validate this hypothesis, we added an additional feature to capture the temporal aspect, i.e., the difference between the guild size in the current snapshot and the previous one. Positive value indicates a growing guild, while negative value indicates a shrinking guild. We have computed this temporal feature for all guild-snapshots. The correlation coefficient

with the class label is -0.0382. The negative correlation is expected. Correlation appears very mild, indicating that past history is not a strong indicator of future trend. Furthermore, including this feature in the feature set for classification does not improve accuracy either. Essentially guild stability can be predicted from the guild features listed above (composition, activity, and structure), and temporal continuity seems to provide little additional information. Table 11 gives the detailed prediction results; it can clearly seen that we are able to achieve high accuracy ($85\%$) in predicting guild stability on the Eitrigg server. Similar analysis is performed on Bleeding Hollow, a player-vs-player (PvP) server, and Cenarion Circle, a role playing (RP) server. We achieve qualitatively similar results for the other two servers; for instance, the random forest classifier produces a prediction accuracy of $81\%$ on Bleeding Hollow, and $84.3\%$ on Cenarion Circle.

## 3.4 Analysis on DBLP Data

### 3.4.1 Dataset

DBLP[2], our second dataset provides bibliographic information on major computer science journals and conferences. Each publication is accompanied by its title, list of authors and conference/journal of publication. For the purposes of our study, we view DBLP as a social network of researchers who co-author papers at different conferences or in different journals. Thus the data resembles the social structure of our WoW dataset; the friendship network is defined by linking people that have co-authored a paper and the conferences/journals serve as groups where these friendships are formed. Table 12 shows the size of our data and the network that we construct from the raw data. An important distinction between the two datasets is the group membership requirement; in the DBLP network an author can often be a member of multiple groups at any given point in time though with varying commitments.

[2] `http://dblp.uni-trier.de`

| Statistic for DBLP | |
|---|---|
| Number of Publications | 1,607,524 |
| Number of Authors | 1,105,457 |
| Number of Conferences/Journals | 7073 |
| Number of Friendship Edges | 7,367,343 |
| Number of Membership Edges | 5,084,657 |

Table 12: Overall Network Statistics for DBLP

### 3.4.2 Gauging Group Stability

As mentioned earlier there is no concept of an author quitting a group in the DBLP dataset; on the contrary an author is a typically a member of several groups. This is a more commonly occuring scenario in most online social networks as compared to the group membership dynamics of World of Warcraft. The DBLP dataset is also used in [12] to study the formation and evolution of groups. Due to the lack of an explicit quitting action most studies have focussed on evolution from the standpoint of growth. We take a different approach when tackling datasets such as DBLP. Even though an author can belong to multiple groups his activities in individual groups can vary significantly over the course of time. Thus we need a measure that can quantify the involvement of a person in a community. We believe such a measure should encapsulate the following properties,

- A person that contributes frequently to a group should have a higher involvement score as opposed to a person that contributes rarely.
- Recent involvement/activities in a community should be weighted higher than past activities in the community.
- The prolificness of the author should be reflected in the measure of involvement.

Since we have timestamped (year of publication) data detailing the activities (publications) of a person (author) in a community (conference/journal), we can define

a measure that reflects all of the above properties by adapting the exponential summarization kernel described in [124]. Let $N_1, N_2 \ldots N_t$ denote the number of publications of an author in a given group at discrete time intervals $t_1, t_2 \ldots t_t$ & $P_{A,t}$ denote the standing/prolificness of the author at time $t$, the "Involvement Score" of the author $A$ in the given group $G$ at time $t$ is defined as,

$$I_{A,G,t} = \begin{cases} (1-\theta) I_{A,G,t-1} + \theta N_t P_{A,t} & \text{if } t > t_0 \\ \theta N_t P_{A,t} & \text{if } t = t_0 \end{cases} \tag{6}$$

where $t_0$ is defined as the initial time and $\theta$ controls the rate of decay. The prolificness of an author can be defined in several ways; we define it as the ratio of the total number of publications the author has at time $t$ to the total number of publications the most prolific author has at time $t$. Prolificness ranges between $[0, 1]$ and serves as a way of determining standing of the author. As mentioned before, the standing can be computed in different ways, for example, total citation count being another effective measure of calculating prolificness. However, the DBLP dataset has no way of determining citation counts, experimenting with other prolificness measures is out of scope of this paper. Table 13 demonstrates the use of the "Involvement Score" measure that we define to uncover trends in publications for prominent authors.

Now that we have defined a measure in Equation 6 to quantify the involvement of a person in a community, we proceed to define "Membership Score" of a group $G$ at time $t$ as the sum of involvement scores for all it's members. Formally,

$$MS_{G,t} = \sum_{A \in G} I_{A,G,t} \tag{7}$$

The membership score that we define has the following desirable properties,

- A group with more number of regularly contributing members has a higher membership score as compared to a group with large number of infrequently active members.
- A group that has more members of repute/standing will have a higher membership score as compared to a group with fewer prolific members.

| Year | Top-3 Conferences | | |
|---|---|---|---|
| 1996 | STOC | FOCS | SODA |
| 1998 | STOC | DM&KD | VLDB |
| 2000 | STOC | FOCS | JComputing |
| 2002 | JCSS | STOC | JACM |
| 2004 | FOCS | JACM | STOC |
| 2006 | FOCS | KDD | IPSN |
| 2008 | JComputing | KDD | EC |
| 2011 | ICWSM | WWW | FOCS |
| 2012 | WWW | ICWSM | WSDM |

Table 13: Top-3 conferences based on Involvement Score for Jon Kleinberg. One can clearly see a change in the trend, from publishing in Theory conferences (yellow) to publishing in Data Mining conferences (green).

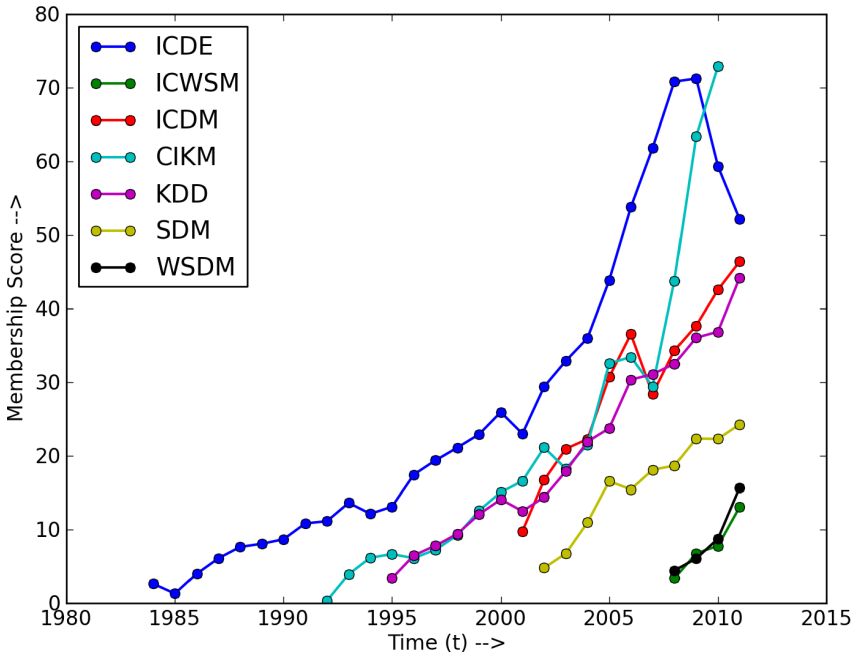



Figure 6: Membership score across time for well-known Data Mining Conferences.

Figure 7 plots the histogram of membership score for groups in the year 2011. It can be seen that most groups have low membership scores with a chosen few

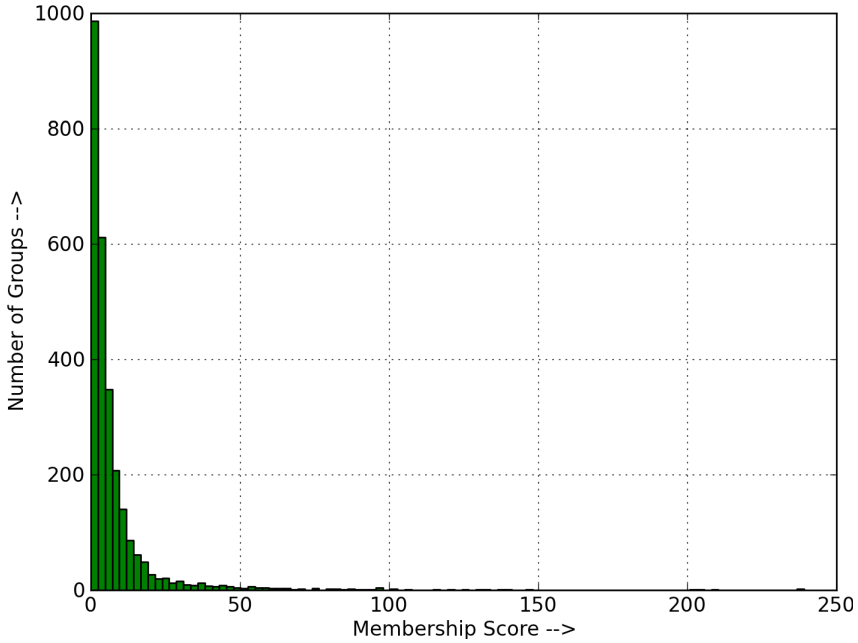



Figure 7: Histogram of membership scores for groups in 2011.

| Conference | Publications | H-index | Membership Score |
|---|---|---|---|
| ICDE | 1303 | 35 | 52.15 |
| KDD | 670 | 30 | 44.20 |
| CIKM | 1348 | 26 | 95.81 |
| ICDM | 1197 | 18 | 46.39 |
| SDM | 338 | 18 | 24.28 |
| ICWSM | 221 | 18 | 13.06 |
| WSDM | 199 | 18 | 15.69 |

Table 14: Top-7 conferences in Data Mining with their Membership Scores

groups (prestigious conferences & journals) exhibiting high membership scores. Figure 6 plots the membership score for 7 well-known conferences in the Data Mining area across the length of their existence. In order to test the efficacy of the membership score, we compute the membership score in 2011 for top-7 conferences in the Data Mining area in the last 5 years (as ranked by H-index [65]

using Microsoft Academic Search[3]). Table 14 shows that the two measures are in rough agreement with each other. It is important to point out that we do NOT intend to advocate the membership score as a replacement for H-index (and such related measures). The membership score is able to capture group dynamics and hence can be used to gauge group stability. We compute the membership score for a group at regular time intervals. Thus computing the percentage increase/decrease in the membership score over the previous interval would then give us an accurate idea of whether a group is stable or shrinking. Given that a group $G$ has membership score of $ms_1$ at time snapshot $t_1$ and membership score of $ms_2$ at time snapshot $t_2$ ($t_1 < t_2$) the percentage change in membership score is defined as $\delta = \frac{ms_2}{ms_1} - 1$. We could then label a group as being in the stable or shrinking phase at time $t_2$ as defined in equation 5. In our experiments we compute the membership scores at yearly intervals (i.e. $t_2 - t_1 = 1yr$).

### 3.4.3 Conference-Level Features

In order to model group stability for the DBLP dataset we consider a range of features that can be broadly classified into three categories: (1) conference/group-specific, (2) publications/activities-specific, and (3) structural features. The following is a list of group specific features,

**Number of members:** The size of the group at time $t$.

**Length of existence (in years):** This feature calculates the number of years since the conference/journal came into existence.

**Membership Score:** Membership Score of group at time $t$ as defined in 7.

**Average Prolificness:** Average Prolificness of group at time $t$, where prolificness is a measure of standing/repute for an individual. It ranges between $[0, 1]$.

We compute the following list of features to capture the activities of members in a particular group,

[3] `http://academic.research.microsoft.com`

**Total & Average Number of Collaborations Within & Outside Group:** These features capture the number of collaborations involving the group members. These collaborations can be within the given group or in some other groups.

**Total & Average Number of Publications Within & Outside Group:** These features capture the number of publications for group members. Again an individual may have publications within and outside the given group.

**Average Member Loyalty Coefficient:** Loyalty Coefficient for a group member is defined as the ratio of the number of publications that member has in the given group to the overall number of publications of the member. It ranges between $[0, 1]$ and is a measure of the loyalty of the member towards a particular group he/she is a member of.

Following is a list of features intended to capture the connectivity information of a group,

**Average clustering coefficient of group members:** We measure the clustering coefficient at each group member node and then calculate the average of this clustering coefficient across all group members.

**Average clustering coefficient of group members within the group:** The clustering coefficient calculated in this case only takes into consideration the graph induced by all members of the guild.

**Entropy of degree distribution:** This feature is a good measure of diversity in node connectivity.

#### 3.4.3.1 Feature Importance & Correlation

We perform similar analysis as performed on the WoW dataset. We compute features along with class labels (1 for shrinking and 0 for stable) at yearly intervals since most conferences/journals have an yearly cycle of publication. Thus, we are trying to predict one year into the future as to whether a group will remain stable or not. This results in around $40,000$ feature samples; $22.51\%$ of these samples have "shrinking" class labels & $77.49\%$ of the samples have "stable" class labels.

| Category | Feature | Correlation Coefficient |
|---|---|---|
| Group | number of group members | **-0.3008** |
| | length of existence | 0.0499 |
| | membership score | **-0.1948** |
| | average prolificness | **-0.5443** |
| Activities | average number of collaborations within group | **-0.6202** |
| | average number of collaborations outside group | **-0.4874** |
| | total number of collaborations within group | **-0.2293** |
| | total number of collaborations outside group | **-0.2687** |
| | average number of publications within group | **-0.7245** |
| | average number of publications outside group | **-0.5216** |
| | total number of publications within group | **-0.2670** |
| | total number of publications outside group | **-0.2732** |
| | average member loyalty coefficient | **-0.6114** |
| Structural | average clustering coefficient | **-0.6477** |
| | average clustering coefficient in group | **-0.6705** |
| | entropy of degree distribution | **-0.7144** |

Table 15: Correlation coefficient between class labels and feature values for DBLP dataset. Correlation coefficients with absolute value exceeding 0.10 are marked in bold-face fonts.

In order to avoid overfitting we draw equal number of samples from both classes. Table 15 reports the correlation coefficient between each feature and the class labels (1 for shrinking, 0 for stable) for the DBLP dataset.

Assessing the importance of features by computing the Information Gain, the top ten features are listed in table 16. Tables 15 and 16 are in general agreement about the important features required for the predictiont task.

| Rank | Feature | Category |
|---|---|---|
| 1 | total number of publications within group | Activity |
| 2 | number of members | Group |
| 3 | total number of collaborations within group | Activity |
| 4 | average prolificness | Group |
| 5 | average number of publications within group | Activity |
| 6 | total number of publications outside group | Activity |
| 7 | average number of collaborations within group | Activity |
| 8 | total number of collaborations outside group | Activity |
| 9 | average member loyalty coefficient | Activity |
| 10 | entropy of degree distribution | Structural |

Table 16: Top ten features, ranked in descending order of information gain. In the category column, G stands for group-specific, A stands for activity features, and S stands for structural.

| Class | Classification Accuracy | Precision | Recall | F-measure |
|---|---|---|---|---|
| Stable | | 0.878 | 0.942 | 0.909 |
| Shrinking | 90.55% | 0.937 | 0.869 | 0.902 |
| Weighted Average | | 0.908 | 0.906 | 0.905 |

Table 17: Group Stability Prediction Results using Bagging

### 3.4.4 Predicting Group Stability

Again, we will use supervised learning techniques and apply them to our feature set to see if we can predict group stability. Due to the unbalanced class problem, we randomly draw equal number of samples from both the classes ($\approx 9000$ samples per class). Table 17 shows the accuracy achieved by using Bagging (we achieve similar accuracy levels by using Decision Trees and Random Forests) . Bagging achieves the best accuracy of $90.55\%$ with a MAE of $0.1402$ and a MSE of $0.2601$; proof of the fact that our feature based approach produces significantly high accuracy in predicting group stability.

## 3.5 Internal Connectedness of Friends

The study of Backstrom et al [12] is amongst the first to comprehensively analyse evolution of groups using real-world social networking data. They demonstrated that the probability of joining increases as the density of linkage increases among the individual's friends in the community. These results are supported by arguments based on social capital [32, 33] that suggest that there is a trust advantage to having friends in a community who know each other. An individual joining such a community is assured of the fact that such a community is a close-knit family of members who know most of the other members.

At the same time Backstrom et al. pointed out that cogent arguments [27, 56] also support the opposite finding; this theory based on weak ties suggested that there is an informational advantage to having loosely connected members. This provides an individual multiple "independent" perspectives; he/she could join based on any one of the ways.

Empirical evidence based on the Live Journal[4] dataset used by Backstrom et al. made them conclude that trust advantage had a stronger effect than informational advantage. Kairam et al. [75] shed further light on the group evolution and growth process. They too touched upon this problem; empirically they came to the same conclusion i.e. probability of joining increases as the density of linkage increases among the individual's friends in the community. They also tried to solve the paradoxical finding of why highly clustered groups tend to have lower growth rates overall. Their findings suggested that some groups grow by appealing to common interests and identities (non-diffusion growth) while other groups grow by virtue of its extra-group connections (diffusion growth). Furthermore they conclude that if a group relies on diffusion growth its scope for growth is limited to the number of ties its members have to non-members. Thus such groups will eventually suffer from lack of new members. Thus, even though high clustering in a group will lead to increased membership it will also lead to diminishing returns (with respect to growth) down the road. In their findings (based on the Ning[5] dataset), they are able to show that groups that grow to small sizes are those that rely on diffusion growth

[4] http://www.livejournal.com

[5] http://www.ning.com

whereas groups that grow to large sizes are those that rely more on non-diffusion growth.

We try to validate the theories and findings put forward in [12, 75] using our datasets as follows,

**WoW Dataset:** We compute the correlation between the features "average clustering coefficient in guild" and "number of new members". The "average clustering coefficient in guild" allows us to quantify the density of linkage amongst a guild's members. The correlation coefficient is $-0.0584$ i.e. weakly negatively correlated which tends to suggest support for informational advantage. This is an interesting finding which hasn't been observed in previous studies. We also compute the correlation between "average clustering coefficient in guild" and "percentage change in number of members over the previous snapshot". This correlation comes in at $-0.00959$ which indicates that density of linkage does not play any role in determining guild growth.

**DBLP Dataset:** Again we compute the correlation between the features "average clustering coefficient in guild" and "number of newly active members". The value of correlation is $0.2530$ which shows support for trust advantage over informational advantage. The correlation between "average clustering coefficient in guild" and "percentage change in membership score over the previous snapshot" turns out to be $0.0061$ indicating again that density of linkage does not play any role in determining guild growth.

Our findings indicate as far as WoW data is concerned, individuals join guilds due to common interests and identities; thus guilds in WoW are characterized by non-diffusion growth. On the other hand in the DBLP data most of the growth can be characterized as diffusion based growth. These findings are also due to the nature of the social networks. WoW is a multiplayer game where individuals work towards an objective of being successful at playing the game. Gamers are likely to join guilds based on common objectives rather than based on trust factors. On the other hand DBLP data is a co-authorship network where edges indicate collaborations at a particular conference or in a given journal. Thus in this case links amongst an individual's friends indicates stronger endorsement for that group from your peers.

## 3.6 Conclusion

Our analysis has shown that it is possible to predict group stability with high accuracy using a range of features that describes the group composition, activities within the group & structural aspects of a group. We have experimented with two large social networking datasets and have been able to achieve similar accuracy levels on both datasets. We have also defined an efficient measure of gauging group membership in scenarios where a person is likely a member of several groups. Our analysis can easily be extended to other online social networks and is also scalable to large networks. The study also shows that it is important to choose features from multiple perspectives, in fact combining diverse features is essential to predictor performance.

# Chapter 4

# Modeling Attrition within an Organization

## 4.1 Introduction

As computer and communication technologies become integrated into people's daily life, many actions and decisions that used to happen in real-world are now carried out on online platforms. For instance, email has become a ubiquitous means of communication, and social groups are increasingly important for people to share information with their real-world friends. The same happens in corporate environment. An increasing number of companies now rely on technologies such as Facebook-like social platforms, twitter-like information sharing platforms, and instant messages for their daily operations. As real-world actions manifest into online data, modeling online behavior in relation to the real-world behavior becomes an interesting and important research topic in both computer science and sociology. Researchers try to address questions such as how people's real-world social context shows up in online social data, whether and how one's action/decision is influenced by online interactions, and whether such manifestation and influence can be modeled and predicted. Many approaches have been proposed and experimented with, for instance, network analysis of social network graphs, content analysis of email text, and temporal analysis of email traffic. The topic of this chapter is along the same line. We study online social interactions, in particular, email communications,

and use them to understand real world behaviors of the involved individuals.

In this study, we focus on the problem of understanding attrition within real-world organizations. Attrition rate, also called churn rate, in its broadest sense, is a measure of the number of individuals or items moving out of a collective over a specific period of time [1]. The term is used in many contexts, for example, referring to users switching to different service providers under a subscriber-based service model, players changing affiliation in online multiplayer games, user involvement and movement in online social networking platforms, and employee turnover within an organization. Attrition rate reflects an important aspect of group stability and thus is highly related to the profitability of a business. Analyzing, modeling, and predicting churn is of great practical significance and has also been extensively studied in research communities. Modeling and predicting attrition in a corporate setting is of particular practical importance.

However, the problem of modeling churn in an organization has received very little or nearly no attention in terms of concrete data analysis. This is largely due to the difficulty in data collection. Privacy concerns are often a major hurdle to data collection and need to be properly addressed. In our work, we have spent extra effort to guard personal information and designed a two-stage approach: (1) careful anonymization in the preprocessing stage to remove personally identifiable information (PII) such as name, address, email address and all numbers, and (2) the innovative design of an email logger, which processes emails and retains only aggregated statistics. Furthermore we assure that the aggregated statistics are sufficiently abstract such that the original content and/or meaning cannot be reconstructed from the feature set. These measures mitigate the privacy concerns and are key to the success of our data collection.

Besides the challenges of data collection, modeling and predicting attrition in a corporate environmetn is difficult from a data analysis perspective. First, any collected online data is inherently incomplete. It is impossible to collect a complete dataset of an individual in his/her social context – whom the individual is in contact with, what he/she is up to, the person's emotional state, etc. Any data collection approach is a best-effort approach. To make things worse, the ground truth, i.e., whether and why an employee is leaving the company, is noisy in nature or

sometimes even lacking. There are various possible reasons attributing to an employee's departure, for instance, a voluntary leave (e.g., better opportunities offered by a competing company, taking time off for personal reasons, etc) or an involuntary one (management administering a lay-off as an organizational cost-cutting measure, or being fired for job disfunction). The exact underlying reason is unobservable. This makes the problem of attrition modeling and prediction difficult. However, we argue that, despite the diversity in possible reasons, almost all of them likely stem from job dissatisfaction (job mismatch, feeling devalued, stress, lack of support, and so on). This has been validated by extensively studied scenarios in social science literature [49, 73].

In this chapter, we present a data-driven approach to attrition modeling and prediction. To the best of our knowledge, this is the first study of its kind to tackle this problem using real-world data. We obtain real workplace activity and communication data from two sources, a small startup company ($43$ employees) and a large US company ($3600+$ employees), as a platform for studying this problem. From the dataset, we extract a rich feature set and first perform correlation analysis to discover which features are strong in correlation with the departure. Some findings are expected and well in alignment with qualitative findings in social science studies. For instance, quitters may be involved more in external communication (i.e., with others outside of the company) and less in internal communications (with colleagues within the company). Quitters may initiate fewer email conversations and instead forward more onto colleagues. Our analysis has also discovered some unanticipated findings. For instance, we originally anticipated that through emails people may convey less positive sentiment and more negative sentiment as they disengage from the company, but the correlation analysis indicates that both negative and positive sentiment go down, and people become generally less expressive. This phenomenon is observed in both datasets. From these correlation observations, we build a model to predict if and when an employee is likely to quit the company. Our predictor achieves a modest accuracy of roughly $60 - 65\%$ prediction accuracy over the dataset from the large company. Giving the noisy nature of the data and the difficulty in attrition prediction, the prediction accuracy is encouraging.

The main contribution of our research is as follows:

1. We have designed a privacy preserving data logging and feature extraction

approach to capture a rich online behavior dataset free of personally identifiable information.

2. Through correlation analysis, our study shows that people's online behavior often exhibits a change point due to quitting intent. Though such change point has been speculated in various social studies, our work is the first to validate its existence from online data traces.
3. Over a large dataset in a corporate environment, we have built a predictive model predicting incipient quitting behavior based on online data with a moderate prediction accuracy.

## 4.2 Related Work

Here we briefly summarize related work on attrition from social science studies, business surveys, and data analysis.

### 4.2.1 Qualitative Findings from Social Science Studies

Social science studies have discovered that company size, industry and pay scales play a key role in determining attrition rate [1]:

- Larger companies tend to have lower rates of attrition.
- Industries that employ a large number of unskilled labor have a higher rate of attrition as compared to the ones that largely require skilled labor.
- Attrition rate is the highest amongst lowest paying jobs and vice-versa.

These findings have provided us with qualitative insights, but have not yet reached the level of mathematical precision that can be deployed to perform churn modeling and prediction. Our work aims to fill this gap.

### 4.2.2 Studies on Churn Rates

In the scenario of subscriber based services, subscribers may leave a service provider for a number of reasons, including customer dissatisfaction, cheaper/better

services/products provided by the competitors, or better marketing of products by competitors. Since this scenario has a direct impact on the profitability of a company, often companies come up with strategies to stem this exodus of subscribers. Strategies such as, creating barriers for subscribers to prevent them from switching or providing incentives such as loyalty programs are utilized for this purpose. Companies go this length because the cost of retaining an existing customer is far less than acquiring a new one [60]. The telecommunications industry gives special attention to this problem [37, 48, 67, 70, 93, 101, 131, 154]. This is due to the low barriers involved in switching service providers. The problem has been studied in other service-based industries as well, such as banking [35], ISPs [68], credit cards [104], insurance [100] and P2P networks [64].

A particular example is the video-gaming industry, a huge and growing market around the world, valued at about $65 billion in 2011 [34]. In order to generate revenues, it is critical that a given game is able to (a) attract new gamers and (b) retain gamers that are already playing the game. Coelln [136] talks about different metrics that can be used to gauge the success of a game in retaining current gamers. The strategies used by various gaming providers in retaining loyal users is discussed in [135]. Various studies [20,79] have also analyzed the problem of churn prediction in online games.

The same problem of reducing churn rates appears in the scenario of social networking platforms. In recent years there has been a proliferation in the kind and number of social networking platforms available to people for interacting in the online space. For a social networking platform to remain active, it is important that it is able to attract users to contribute over a long period of time. Thus identifying people who are likely to churn early would allow the platform to investigate such users and make changes (or add features) to rectify such a situation. In [41, 77], the authors try to identify user features that can lead to churn in social networks. They then use these features to train classifiers for churn prediction achieving fair accuracy.

Most of the work listed above considered customers leaving/quitting the services as separated, independent events. They seek reasons from the internal of an individual. The proposed method for lowering churn rate is either by improving user satisfactory and providing better incentives, or by raising the bar of switching

services and imposing penalties. Nevertheless, in reality customers do not make quitting decisions independently. There is a hidden, yet powerful social factor behind user decision-making processes. It is often the case that a customer decides to subscribe/unsubscribe a service based on the decisions of his/her social friends. The aspect of social relationships in modeling and predicting churns is only studied until very recently. In our previous work [114], we looked at a related problem of predicting departures from groups and whether such departures are likely to cause damage to a social group. We performed the analysis using World of Warcraft (WoW), the most popular online role-playing game, as a platform. The analyses from real-world data sets indeed demonstrate a clear social influence when people make decisions on joining/quitting social groups.

### 4.2.3 Career Switch Modeling

A recent paper by Wang *et al.* [137] proposes a probabilistic model for career switching, to help a career recommender system in providing recommendations at the right time, i.e., to identify the time-period when a user is likely to be susceptible to making a career-switch. It models the duration between two successive job-related actions (such as a promotion or a churn event) using a proportional hazard model [36]. Proportional hazard is a technique that originates from reliability theory, describing the life span of a component (in this case the length that the user remains at the same job level) as a function of a baseline duration modulated by exponentials of a set of related factors. The paper then fits the proportional hazard model to a job application database from LinkedIn[1].

Our work is fundamentally different from [137]. For instance [137] models using a survival model on tenure, i.e., for how long is an individual expected to remain in the same position, while our approach examines data traces and discovers which data features may point towards a manifestation of an underlying attrition. Unlike [137] that uses tenure to decide when to pop job recommendation to users, our method predicts incipient departure based on features that may capture precursors of one's departure.

[1] `http://www.linkedin.com`

## 4.3 Analysis on Startup Email Dataset

### 4.3.1 Dataset

For attrition modeling, we recruited 43 participants from a research lab to share their social interaction data. The participants have diverse job roles: mangers, individual contributors, and administration staff. About half of the participants were associated with an internal spin-off startup company, which had not been successful in its business venture and later got incorporated back into the research lab. During the business turmoil, a significant portion of the employees left the company. From a supervised learning perspective this is an ideal dataset for analyzing quitting behavior, with positive and negative data samples and ground truth of who quit the company and when.

As we mentioned earlier in Section 4.1, one major barrier to attrition analysis is data collection. We have designed an email feature extraction tool that respect users' privacy. Our tool can be deployed as a software agent installed on participant's PC to extract features from Outlook emails. Our tool takes two steps to protect privacy: (1) Participants are assigned a random ID, and only this ID is used to identify the participant, and all PII such as name, address, email address, and all numbers are ignored from email content. (2) Email content is processed to extract aggregated features such as word frequency counts. No raw content is logged in a feature set, and hence the original content cannot be reconstructed from the feature set. Furthermore, upon completion of feature extraction, the software agent uploads only aggregated features such as word frequencies onto an encrypted server and uninstalls itself.

For PII removal, emails go through a set of carefully designed pre-processing steps. First, reply lines and signature blocks are detected using regular expression and conditional random field techniques, as describe in [29]. The identification of reply lines and signature blocks serves two purposes. First, the reply lines facilitate email thread reconstruction, e.g., $A$ sends an email to $B$, which $B$ then replies to $A$ and cc to $C$. By capturing email threads we can formulate an email graph and analyze its structure. Secondly, reply lines and signature blocks often contain personal information and sometimes redundant or irrelevant content. For instance, reply lines often quote original text from a previous message, and signature blocks

sometimes have quotes from famous people. We remove these content portions when extracting content features.

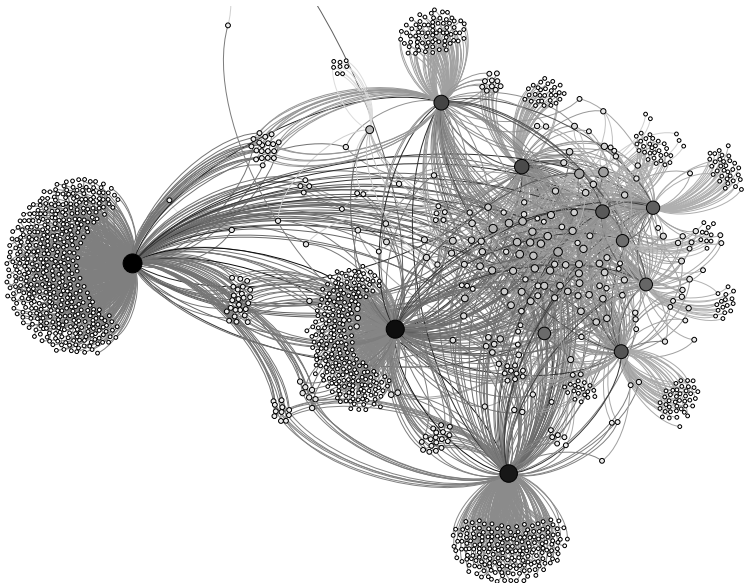


Figure 8: Startup Email Graph. The larger, darker dots indicate internal employees, whereas the smaller, lighter dots indicate external email addresses.

### 4.3.2 Feature Set

Loosely speaking, we extract three categories of features: (1) meta-features, (2) graph structure features, and (3) content features.

Meta-features summarize various aspects of email usage: when an email send/receive event occurs, how many emails one sends a day, how many friends (defined as distinct people with email correspondence) one may have, how many internal (to the company) emails/friends vs. external, and how many email correspondences are done outside of regular work hours.

Graph structure features reflect topological characteristics of email communication. All participants' email archives are used collectively to generate a dynamic email graph, where nodes are the participants, and edges are emails. Figure 8 visualizes the email graph that emerges from the communication traces. It is of interest to analyze the structure features of any given node, such as its degree or weighted

degree, how tightly a node's local neighborhood is connected, and how balanced or skewed one's communication is within its neighborhood.

Content features are computed using text analysis techniques described in [125]. Here we briefly summarize the features:

**Word statistics:** This includes bag-of-words frequency counts for common words and the frequency counts for part-of-speech (POS) tags (e.g., noun, verb and adjective, superlatives, etc).

**Sentiment features:** A word can be positive, negative, neutral, or both positive and negative. The frequency for each category is counted. Negations are handled with special care.

**Writing style:** Professional emails often have a conventional structured format, with greeting in the beginning and closings at the end. Personal and professional emails often use smileys to express emotions. We add these to the feature set to investigate the expressiveness.

**Speech act scores:** One important use of work-related emails is to request or prompt certain actions, such as negotiation or task delegation. We use a pre-trained speech predictor [31] to predict six Speech Act scores: request, deliver, commit, propose, meet, and communicate data. Although not extremely accurate (around 70% F1 scores), the Speech Act scores make a good feature set due to its semantic importance, especially in work-related content.

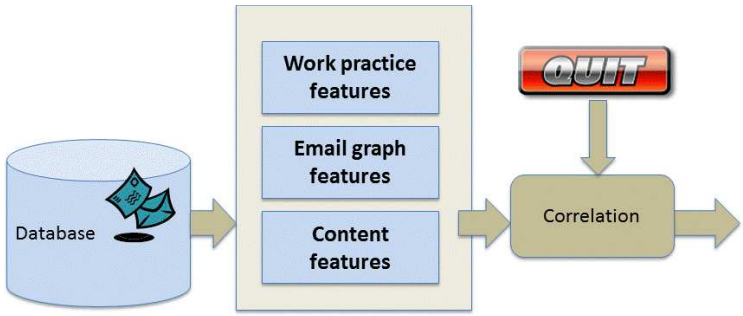



Figure 9: Quitting dynamics analysis from email features for the Startup Dataset.

Figure 9 outlines the methodology for our analysis. From the dataset, meta-features, graph structure features, and content features are computed and collated

into a feature set. They are then correlated with quitting ground truth. Due to the limited population size (43 participants total), we cannot train a reliable predictive model, but nevertheless it is important to address questions such as whether and how a quitting decision manifests into observation features, and which features are informative and indicative of quitting.

### 4.3.3 Class Labels

For investigating quitting dynamics, we first associate labels to differentiate multiple stages of one's employment. Given a participant, we take his/her employment period and segment it into 4 segments:

**A warm-up period (class label 0):** the first 6 weeks of employment is labeled as the warming up period. Email events in this period are discarded because they are not representative of one's regular work. Rather, the participant is communicating to familiarize with his/her environment.

**An exit period (class label 3):** the last 3 weeks of employment is labeled as the exit period. The employee has already made a decision to quit the company. The activities during this phase are usually wrapping up existing work and hand-off to colleagues.

**A first and second working period (class labels 1 & 2):** Cutting out the warm-up and exit periods, the rest of one's employment is divided into two equal-length segments. It is of interest to see whether one's email behavior changes during these two working periods (transition $1 \rightarrow 2$) and as he/she starts to exit (transition $2 \rightarrow 3$).

### 4.3.4 Correlation Analysis

Table 18 reports email features with non-negligible correlation coefficient with the employment stage labels. The second column ("transition $1 \rightarrow 2$") lists the correlation coefficient as participant transits from his/her first employment period to the second. The third column ("transition $2 \rightarrow 3$") lists the correlation coefficient as participant transits from the second employment period to the exit period. A

positive correlation coefficient indicates that a general increase of feature value as participant progresses through the employment periods, while a negative correlation indicates a generally decreasing feature value.

| Class | Feature | Transition $1 \rightarrow 2$ | Transition $2 \rightarrow 3$ | Change Point |
|---|---|---|---|---|
| Meta-features | number of emails | 0.279 | -0.252 | Fewer emails and internal communications |
| | number of distinct recipients | 0.318 | -0.199 | |
| | number of emails internally sent | 0.279 | -0.252 | |
| | number of internal recipients | 0.391 | -0.199 | |
| | percentage of internal recipients | 0.109 | -0.074 | |
| | number of after hours emails | 0.118 | -0.040 | Change in work habits, fewer after-hours activity |
| | number of attachments | 0.031 | -0.101 | |
| | average number of TO recipients | 0.126 | -0.069 | Fewer composed or replied emails, increased forwarded emails |
| | average number of CC recipients | 0.318 | -0.065 | |
| | number of forwarded emails | 0.155 | -0.017 | |
| Graph Structure | entropy over email recipients | 0.284 | -0.146 | More skewed communication, fewer cliques |
| | clustering coefficient | 0.127 | -0.228 | |
| | weighted degree | 0.333 | -0.108 | |
| Content | number of exclaimation marks | 0.117 | -0.054 | Reduced expressiveness in communication |
| | number of emails with positive polarity | 0.257 | -0.244 | |
| | number of emails with negative polarity | 0.279 | -0.225 | |
| | number of emails with rare/complex words | 0.253 | -0.258 | |
| | number of question marks | 0.027 | -0.085 | No apparent changes |
| | number of POS Tags JJS (Superlatives) | -0.061 | -0.046 | |

Table 18: Correlation between email features and ground truth. The column "transition $1 \rightarrow 2$" lists the correlation coefficient as participant transits from first employment period to the second. The column "transition $2 \rightarrow 3$" lists the correlation coefficient as participant transits from the second employment period to the exit period. Correlation coefficients with significant amplitude ($> 0.05$) are highlighted as follows, red for negative correlation, and blue for positive correlation.

The correlation coefficient for the meta-features exhibits a very distinct change pattern. For instance, as participants transit from stage 1 to 2, the number of emails generally goes up, but as participants transit from stage 2 to 3, the number of emails drops. Same pattern persists for a number of other features. In general people's internal email activities (internal with respect to the company) tend to drop, with fewer emails and fewer recipients. Also fewer activities after the normal working hours are observed. Fewer emails with attachments are observed. All these indicate generally less job-related activities. Furthermore, the number of forward emails

drops but to a much lesser extent than the number of all emails. Percentage-wise the quitting participant is doing more email forwards and less originals or replies. This is also anticipated as people disengaging themselves from work. The email forwards are probably delegation and handoff to colleagues.

Among the feature set for graph structure, the entropy feature is computed over the empirical distribution of email frequency across recipients. It measures how evenly or skewed the participant communicates with email recipients. It takes a high value if the participant communicates with friends evenly, and a low value if his/her email activities are confined to a small subset of recipients. Table 18 shows that entropy drops, indicating that the communication is more skewed to a smaller subset. This is not surprising as we anticipate quitters to communicate more with close friends and less to the broader set of colleagues. Clustering coefficient [142] is defined as the ratio of triangle cliques over the number of possible triplets. It takes value 1 if the participant's local neighborhood is fully connected and 0 if the participants live in a star topology. Table 18 shows that the clustering coefficient generally drops. This is probably because the participant is involved in more external and less internal communication. The internal communication community is expected to be tightly knit, hence closer to a fully connected graph, while the external network is only connected through the participant and hence exhibits more of a star topology.

Analysis over content features turned out to be somewhat surprising. We had originally anticipated that positive sentiment may go down and negative sentiment may go up as people plans to exit. Data seems to suggest that both positive and negative sentiments go down. In addition, exclamation marks and rare/complex words are used less often than before. In general people become less expressive and distant themselves from controversial expressions. On the other hand there is no significant change observed in terms of questioning and the frequency of using superlatives.

Although a change point has long been speculated and empirically observed in social science studies, our work is the first to report its existence in real-world email data. These preliminary results encourage us to look into the possibility of constructing predictive models to further analyze quitting behavior in larger datasets.

## 4.4 Analysis on Large Company Dataset

### 4.4.1 Dataset

The second dataset comes from a large US company with several tens of thousands of employees (hereafter we will refer to this dataset as the "Large Company Dataset"). The company has adopted a logging mechanism for their employees, logging activities on company operated PCs. Some of the types of activities logged are email communications, logon activities, file-access activities, websites visited from work PC etc. For our problem, email communications serve as the most important source of information as compared to other activity data.

| Statistic for Large Company Dataset | |
|---|---|
| Number of Target Internal Employees | 3,615 |
| Number of Other Internal Employees | 23,672 |
| Number of External People | 86,240 |
| Number of Email Communications | 37,619,622 |
| Number of Quitters Identified | 566 (15.65%) |
| Time Range | 14-May-2012 to 01-Oct-2012 |

Table 19: Data Statistics for Large Company Dataset

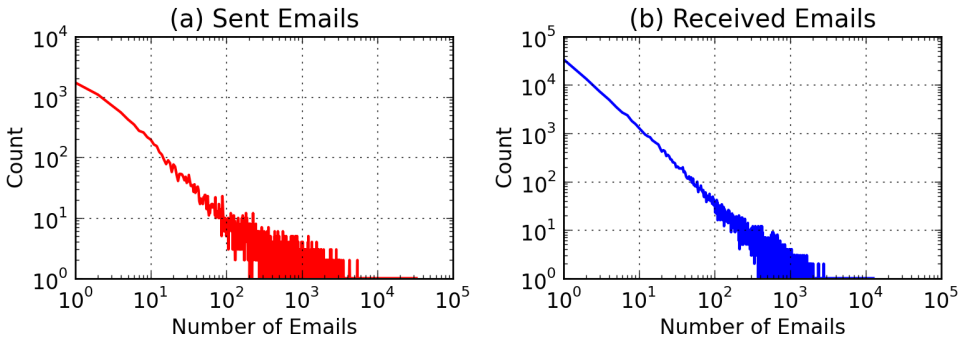



Figure 10: Degree Distribution for Email social graph constructed from Large Company Dataset. The plots shows a power-law degree distribution for Sent (top) & Received (bottom) emails.

While studying our problem we limit our attention to communication data for

a large subset ($3600+$) of employees during a 20-week time period. Table 19 details the scale of the dataset. For analysis, we distinguish individuals into three categories. The first category includes individuals that belong to our targeted subset of employees, i.e., the $\approx 3600$ employees in our dataset. These are the samples for our analysis. The second category are the ones that are employees of the company but not in our subject dataset. They are excluded for purposes of churn analysis. The last category are people external to the company. All communication between the first two categories of people are treated as "internal" communication and all other communication is treated as "external" communication. For the purposes of this analysis we have focused on email communication trace data without using actual email content for analysis. We exclude actual email content from our analysis due to privacy concerns. Since the dataset does not have actual information on employees quitting the company (i.e. date & reason of departure), we will use a simple but effective heuristic to determine the approximate date of departure. We will also model the email communication data as a social network that can then be used to extract certain structural features that could be useful for churn prediction. We construct the social network by placing an edge between two people that have participated in an email communication. Figure 10 plots the degree distributions of the social network constructed from the email dataset.

### 4.4.2 Identifying Quitters

The dataset does not have the ground truth with respect to employees quitting the company (that is treated as proprietary information). Since we require ground truth in order to train and validate our approach, we need to be able to deduce the fact that an employee has quit the company from the given communication and activity data.

We use a heuristic based on "Sent Emails". For a given employee we keep track of his/her sent emails at regular intervals of time (daily in our experiments). We can then classify an employee as having quit the company if we fail to see any emails being sent by this employee for a prolonged period of time such that this employee never sends out any further emails. The "Sent Email" heuristic should allow us to get reasonable ground truth regarding quitters. In our experiments we

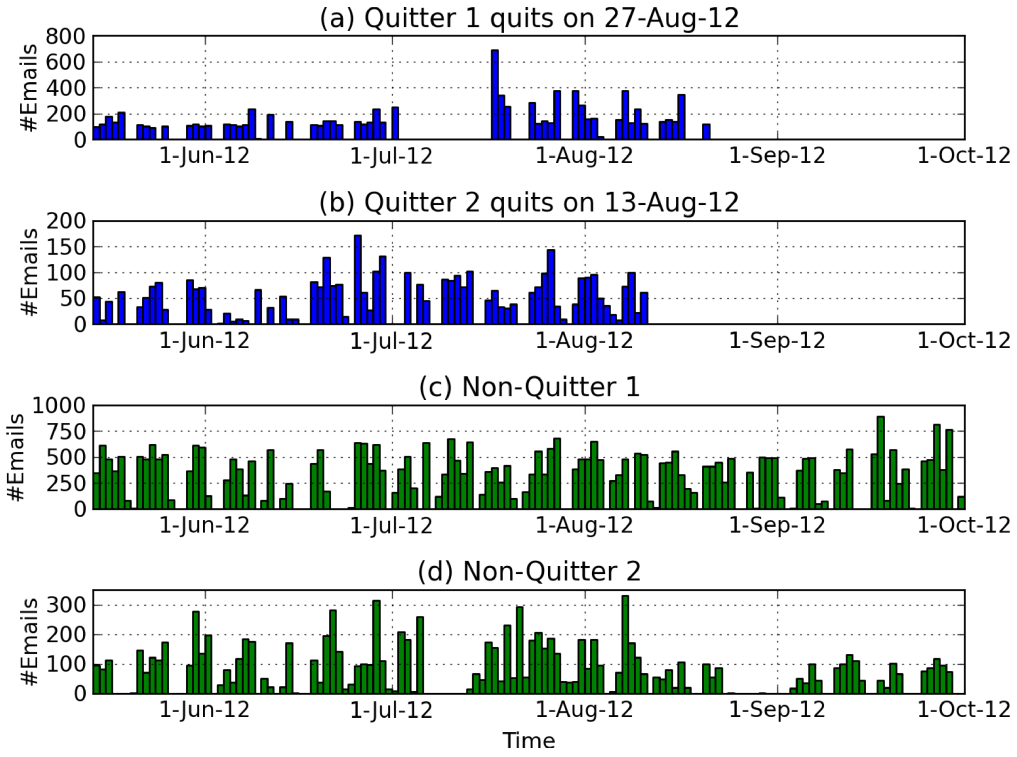



Figure 11: Email Communication Plot for 4 employees. Top-2 employees (blue bars) are those that have been deemed to have quit the company based on the "Sent Email" heuristic". The 2-day gaps indicate lack of email activity on weekends. One can also see a temporary week-long absence for couple of the employees indicating vacation, sick leave etc.

will say that an employee $E$ has quit the company at time $T$ if he has sent his last email at time $T$, followed by complete & permanent absence for at least 21 days. We set the minimum time duration for lack of sent emails as 21 days; that should take care of scenarios where employees are temporarily absent due to vacation, sick leave etc. While we realize that no heuristic would be perfect, we believe that our heuristic is effective in identifying most if not all quitters. We also believe that we can deduce the approximate date of departure for such employees by using the "Sent Email" heuristic. A drawback of our approach would be in cases where employees use (or switch to) machines that have not been registered for logging email and other activities. In such cases we might identify some employees as quitters even though

they might still be employees. We consider such cases as noise; another reason that makes this problem harder than in other scenarios. Figure 11 shows the sent email activity plot for 4 employees, top 2 employees are deemed to have quit the company based on the "Sent Email" heuristic. In all, we identify 566 employees that have quit the company in the 20-week time period in the Large Company Dataset.

### 4.4.3 Feature Set

The first step towards churn analysis entails extracting a rich set of features from the communication dataset. Table 20 provides a list of features that we compute from the dataset. Most features are similar to those described in Section 4.3, except that Meta-features are further split into two categories:

**Email Features:** The features are extracted from email communication data and form the bedrock of our analysis. Email communications data provide the a large set of features that can be used to detect changes in workplace behavior.

**Social Features:** These are features that are a direct byproduct of constructing the email social graph. These features are useful in capturing statistics on one's ego network.

### 4.4.4 Feature Importance & Correlation

The observation dataset is organized into temporal snapshots, sampled at weekly intervals i.e. we compute the set of features for every employee at weekly intervals. Furthermore, we label each feature sample for an employee as being a "quitting" or "not quitting" sample. If an employee is to quit the company within the next 3 weeks, we classify the feature sample for that employee at that particular time as "quitting", if not then the sample is classified as "not quitting". Labeling in such a manner allows us to train classifiers for the churn prediction problem. Also, it allows the classifiers to uncover any precursors that lead to an employee departure. Overall there are about 50,000 feature samples, out of which about 1713 (3.42%) are samples with "quitting" label.

Table 20 reports the Spearman's Rank Correlation Coefficient between each

| Category | Feature | Correlation Coefficient |
|---|---|---|
| Email | percentage of emails sent | 0.0306 |
| | percentage of distinct recipients | 0.0294 |
| | percentage of external emails | 0.0263 |
| | percentage of after hours external emails | 0.0197 |
| | percentage of after hours internal emails | -0.0197 |
| | percentage of internal emails | -0.0263 |
| | percentage of distinct senders | -0.0294 |
| | percentage of emails received | -0.0306 |
| | number of emails received | -0.0419 |
| | number of distinct senders | -0.0426 |
| | number of after hours internal emails | -0.0449 |
| | number of after hours external emails | -0.0541 |
| | number of after hours emails | -0.0566 |
| | number of internal emails | -0.0583 |
| | number of distinct recipients | -0.0663 |
| | number of external emails | -0.0667 |
| | number of emails sent | -0.0670 |
| | number of emails | -0.0703 |
| Social | percentage of external friends | 0.0070 |
| | percentage of internal friends | -0.0070 |
| | number of friends emailed excessively | -0.0314 |
| | number of emails per internal friend | -0.0490 |
| | number of emails per external friend | -0.0542 |
| | number of emails per friend | -0.0569 |
| | number of internal friends | -0.0804 |
| | number of external friends | -0.0920 |
| | number of friends | -0.0930 |
| Structural | clustering coefficient | -0.0453 |
| | email entropy | -0.0695 |
| | weighted degree | -0.0706 |

Table 20: Spearman's Rank Correlation coefficient between class labels and feature values. Correlation coefficients with significant amplitude ($> 0.05$) are highlighted in colors, red for negative correlation, and blue for positive correlation.

feature and the class labels (1 for quitting, 0 for not quitting). We prefer Spearman's correlation coefficient over Pearson's correlation coefficient to mitigate issues that may arise due to outliers and skewed nature of the data. Following are the observations from the correlation analysis,

- Some of the features exhibit distinct trends.
- An employee that is about to quit the company will participate in fewer emails (number of emails sent, number of emails received).
- He/she will communicate with a selected (fewer) group of people (number of friends, number of external friends, number of internal friends, email entropy, weigted degree, clustering coefficient).
- There is a marginal increase in his/her percentage of external emails i.e. emails sent to people outside the company.
- Overall such an employee is likely to be less communicative as compared to other employees.

Table 21: Top ten features, ranked in descending order of information gain.

| Rank | Feature | Category |
|---|---|---|
| 1 | number of external friends | Social |
| 2 | number of friends | Social |
| 3 | number of emails sent | Email |
| 4 | number of distinct recipients | Email |
| 5 | number of emails | Email |
| 6 | number of internal friends | Social |
| 7 | weighted degree | Structural |
| 8 | number of external emails | Email |
| 9 | email entropy | Structural |
| 10 | number of emails per friend | Social |

Table 21 shows the top-10 features as ranked by the information gain criterion. This indicates how informative a given feature is with respect to deducing the class

label (quitting or not quitting). This is consistent with Table 20 regarding which features are most important for churn prediction.

### 4.4.5 Predicting Quitters

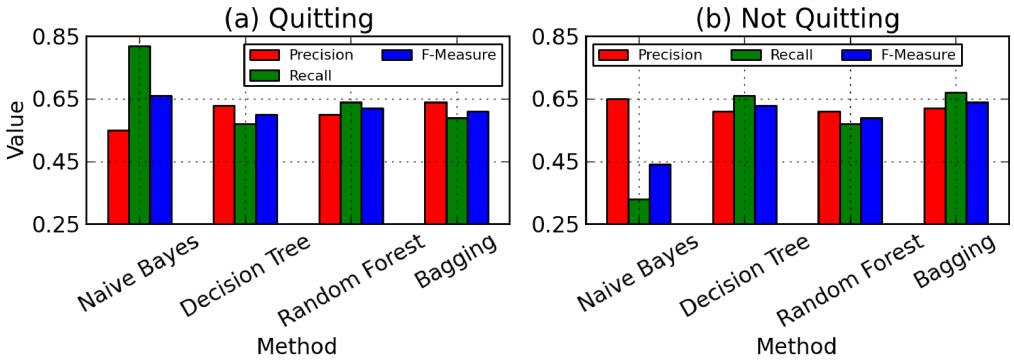



Figure 12: Prediction Results using different classification techniques. The figures plot precision, recall & f-measure for the "Quitting" (left) & "Not-Quitting" (right) classes. The classification accuracy ranges between $58-63\%$ for the different techniques, with Bagging proving to be marginally better than the remaining methods.

| Class | Classification Accuracy | Precision | Recall | F-measure |
|---|---|---|---|---|
| Quitting | | 0.641 | 0.588 | 0.613 |
| Not Quitting | 62.96% | 0.62 | 0.672 | 0.645 |
| Weighted Average | | 0.631 | 0.63 | 0.629 |

Table 22: Prediction Accuracy using Bagging

Given the feature samples along with the class labels, we train classifiers for the churn prediction problem. Due to the skewed nature of the data (i.e. very few "quitting" samples), we first randomly select equal number of samples from both classes. We conduct experiments using a wide range of supervised learning techniques, with Bagging providing the best accuracy. We are able to achieve accuracy levels that range between 58-63% using various classifiers such as Naive Bayes, Decision Trees, Random Forests & Bagging. Figure 12 provides the precision, recall & f-measure numbers for each of the classifiers. Table 22 shows the detailed

results obtained by using Bagging. One can see that we are able to achieve a modest accuracy of about 63% when predicting churn in an organization. The results also indicate that predicting churn in an organization seems to be a much harder problem than predicting churn in other well studied scenarios.

## 4.5 Conclusion

Through analysis on the startup email dataset and the large company dataset, we are able to make the first (albeit modest) headway into the problem of churn modeling prediction in an organization. Though the results are still preliminary, our analysis establishes a data-driven correlation model between people's online behavior and their real-world company churn events. We have identified informative features and found weak trends, indicating that employees become less communicative and less expressive before they are likely to quit the company. Our analysis also has verified the existence of a change point in people's work-related emailing behavior. Furthermore, through analysis on the large company dataset, we can use the weak trends to predict incipient departure with a moderate accuracy.

This effort is inscribed in a broader scope of research seeking to establish correspondence between people's real-world context and online social behavior. We hope that the method presented in this paper can be extended to other similar problems as well. We plan to look into other social dynamics, such as influence propagation, dynamics between work place collaborators, and roles people take in their online social groups. Through such effort we hope to achieve a better understanding of the social space around us, so as to make online platforms more helpful to real-world users.

# Chapter 5

# Quantifying Social Influence in Epinions: A Case Study

## 5.1 Introduction

Consumer ratings, reports, surveys and polls existed long before the age of the Internet[1]. They are revitalized due to the advent of the Internet and numerous eCommerce websites. Consumer ratings have become much more diverse, popular and accessible. While it was not hard to get reports and ratings about a new vehicle before the age of Internet, it might have been hard to find ratings about a toothbrush, a searing pan or a local restaurant.

Many eCommerce platforms such as Amazon and Ebay actively engage their users in providing their opinions or ratings of the products they have bought. Consumer review websites such as Epinions, Yelp, Angie's List[2], etc allow users to review products and also rate reviews written by other users. These websites have come to play an important role in guiding people's opinions on products and in many cases also influence people's decisions in buying or not buying the product. As shown in the global Nielsen survey of $26,486$ Internet users in $47$ markets, consumer recommendations are the most credible form of advertising among $78\%$ of

[1]The *Consumer Reports* magazine is in press since 1936.

[2]The websites are `http://www.amazon.com`, `http://www.ebay.com`, `http://www.epinions.com/`, `http://www.yelp.com/`, and `http://www.angieslist.com`.

the study's respondents [2]. A recommendation from a trusted source goes a long way in shaping others' opinions.

Some websites incorporated social network structures into their rating systems. Users can rate each other as trusted or distrusted relationships. In such environments, we are dealing with two types of data: "rating system" data and "social structure" data. The incorporation of the social network structure into the rating system leads to an interplay between the two that affects both of them. Users may rate each other as trusted/distrusted based on reviews they write. The social structure might influence users in rating the reviews of others. Furthermore, the reviews in the system and the trust/distrust relationships based on those reviews might influence the formation of new relationships.

Our course of study in this paper can be summarized as follows: (1) We investigate whether there is correlation between the opinions of a user's current trustees/friends and formation of his/her future relationships; (2) We also investigate whether there is correlation between the opinions of a user $A$'s current trustees/friends regarding another user $B$ and the score of rating that $A$ would assign to reviews written by $B$; and (3) We then use our findings in building a predictor model that is able to predict with good accuracy the rating a user is likely to assign to a review.

To carry out such an investigation we use data from a widely used consumer review website Epinions [94, 95]. Users write reviews on products or services that may earn them money or recognition. Users also rate reviews written by other users and can also express trust or distrust towards other users. While trust relationships are publicly visible, distrust relations are not visible (except to the user making the relationship). When rating reviews, users have an option of making their rating public or keeping it private.

The trust and distrust relationships for a user are combined to determine the ***web of trust*** for that user; the user is shown reviews written by people within this web of trust more prominently as opposed to other reviews.

### 5.1.1 Problems and Results

We define the problems in the context of the Epinions as follows,

1. We investigate the existence of a correlation between the opinions of a user's current trustees/friends and formation of his/her future relationships. Specifically, if user $A$'s friends collectively have an opinion (trust/distrust) about user $B$, would user $A$'s future relation with $B$ have a correlation with the collective opinion of his/her friends regarding $B$? For example, if more of $A$'s friends tend to *trust* $B$ (rather than *distrust* $B$), would $A$ be more likely to trust $B$? In how many cases would $A$ make a decision in contrast to what his friends think of $B$? How are the choices made by $A$ related to trustworthiness of both $A$ and $B$?[3] To carry out such an investigation, we must be very careful about issues like innate differences in ***trustworthiness*** of different users.

   **Results:** Our findings suggest that there is a strong ***alignment*** between the collective opinion of a user's friends and the formation of his/her future relationships, after we factor out the innate biases of trustworthiness of different users.

2. We also investigate the existence of a correlation between the opinions of a user $A$'s current trustees/friends regarding another user $B$ and the score of rating that $A$ would assign to reviews written by $B$. We are specifically interested in the case where $A$ and $B$ do not have a direct trust/distrust relationship but rather an indirect one ($A$ is not friend/foe of $B$, but one of $A$'s friends/foes is a friend/foe of $B$). We refer to this problem as studying ***friend-of-friend dynamics***. If there is no correlation between friend-of-friend dynamics and a user's ratings, then we can say that the social structure does NOT provide any advantage to the ecology of the rating system. On the other hand, if there exists such a correlation, we could use this finding to recommend better content to the user. It would also imply that the social structure supports or improves the overall user experience by helping him/her identify relevant content. Again, we take into consideration the innate differences in ***rating habits*** of different users.

   **Results:** Our analysis leads us to conclude that in cases where user $A$'s

[3]While Leskovec *et al.* in [86] have looked at triadic closure: i.e. $A$'s relationship with $B$ depending on both $A$'s and $B$'s relationship with an additional nodes $X$, we take an aggregate view of the relationship formation process; i.e. we intend to understand the collective role played by all of $A$'s friends in guiding his future relations.

friends have expressed approval or disapproval of another user $B$, there exists an alignment between the $A$'s rating of $B$'s reviews and his friends opinions regarding $B$. On the other hand, approval or disapproval expressed by foes seem to have no correlation with user's rating whatsoever.

3. We use the FoF dynamics information in the form of features to build a predictor of ratings. We apply the predictor on the ratings assigned by a user to another user's review and gauge its accuracy. The model that we develop starts from raw Epinions data and extracts a diverse set of features. These features help us in predicting the rating a user is likely to assign to a review. Our predictor model achieves an accuracy in excess of $76\%$ with a ROC area (AUC) of $0.91$.

## 5.2 Background and Related Work

The interplay of eCommerce and social networks has been studied in a number of prior work. A bulk of the research has been devoted to identifying, propagating and predicting trust (and distrust in some cases) relationships. Incidentally, Epinions has served as a platform for many of the studies conducted in this domain. In the following we briefly review the relevant work and point out the differences with what is presented in this paper.

### 5.2.1 Signed Relationships

A relationship between two individuals can be signed, i.e., positive or negative. A positive relationship means friends or a trusted relationship. A negative relationship means foes or a distrusted relationship. The study of signed relationships in social network can be traced back to studies in social psychology. The first of these theories known as the *structural balance theory* was formulated by Heider in 1946 [63] and generalized by Cartwright & Harary in 1956 [28]. The theory considers the possible ways in which triangles on three individuals can be signed and states that triangles with exactly odd (one & three) number of positive signs are more prevalent in real networks as opposed to triangles with even (zero & two) number of positive signs. In other words, a triangle with three positive edges means 'the friend of your

friend is your friend'. A triangle with one positive edge and two negative edges means 'the enemy of your enemy is your friend'. Both are commonly observed in real networks while the other two cases with three negative edges or exactly one negative edge are not likely. This is also termed the *strong structural balance*. Davis's theory on *weak structural balance* [38] is a variant of the structural balance theory that states that all types of triangles except the ones with exactly two positive signs are plausible in real networks. Signed relationships and structural balance theory have been widely used in psychology and international relationships.

Leskovec *et al.* [86] studied a number of data sets from online review networks to see if they are consistent with the structural balance theory. However, one must notice that in Epinions the relationships are directional, in the sense that one user rates another user as trusted or distrusted, while the structural balance theory only considers mutual relationships. By ignoring the directed nature of links in their datasets, the authors find alignment with Davis's theory on weak structural balance; their findings though are at odds with Heider's structural balance theory in two out of the three datasets. On the other hand, when considering the directed nature of the trust/distrust links the authors show that strucutural balance theory is inadequate in explaining the online structures that come to develop in the datasets. Instead the authors develop the *status theory* which states that both the sign and direction of the link convey a notion of status. For example a positively directed link is interpreted as indicating that the creator of the link is conferring higher status to the recipient. On the other hand a negatively directed link would indicate that the creator views the recipient as having a lower status. The authors are able to show that the *theory of status* is a better fit to describe the structures that develop in the real-world datasets. In their follow up work on signed link prediction [85], the authors further shed light on these two theories; they find evidence that the theory on balance operates at a local level rather than the global level whereas the theory on status is shown to operate at a local level as well as lead to an approximate global ordering on nodes.

We do not intend to examine the alignment or disalignment of the data sets with previous theories. Instead, we aim to see whether there exists a correlation between the signed relationships and how users rate each other's reviews/opinions.

### 5.2.2 Predicting Signed Relationships

In real data sets it is often the case that only some of the edges are explicitly labeled with a sign. Many other edges may be implicitly signed but not demonstrated in the collected data sets. Thus one interesting problem studied in the literature is to predict the sign of any two individuals.

The first set of papers consider propagation of trust or distrust in a network, thus predict the sign of an edge from the signs of other (neighboring) edges. [57] is amongst the first studies to study the problem of propagation of both trust and distrust in online eCommerce websites. They build on the "path-algebra" model of trust propagation put forward by Richardson *et al.* [121]. Given a small number of expressed trusts/distrusts per person, they are able to predict trust/distrust between any two individuals with good accuracy of about 85%. Predicting trust/distrust between two arbitrary people allows the system to filter/recommend content more effectively based on their inferred likes/dislikes.

[95, 106] build further on this idea. In [95], Massa & Avesani, experimenting with the same Epinions dataset that we intend to use, come to conclusion that local trust metrics computed via trust propagation are more effective in recommending "suitable" content as compared to global trust metrics and collaborative filtering techniques. In an earlier study by Massa & Avesani [94], they show the efficacy of local trust metrics over global trust metrics for controversial users (i.e. users who are liked and disliked in equal measure or users displaying a polarizing streak). Similarly, O'Donovan & Smyth in [106] demonstrate the usefulness of incorporating trust values into a collaborative filtering algorithm. They experiment with the MovieLens dataset [120] and are able to get a 22% improvement in prediction error rates by using trust values. We refer the reader to [55, 76, 111, 117] for works on propagating trust in other related settings.

A separate line of research considered a diverse set of features, at both a network level and an individual level, for "signed" link prediction in online social networks. Liu *et al.* in [89] construct a diverse set of features from both user attributes and user interactions to predict "trust" links in Epinions. They find that adding interaction features improves accuracy of prediction as opposed to using user features alone for predicting "trust". A drawback of their study lies in the fact that they do not use "distrust" link information and thus are able to only predict "trust" or the

lack of it in their dataset. On the other hand Leskovec *et al.* [85] expand the horizon by being able to predict both positive and negative links in online social networks. The experiments are conducted on datasets from Epinions, Slashdot & Wikipedia used in previous studies [26, 57, 83]. They demonstrate high accuracy (in excess of 90% for 2 out the 3 datasets) in predicting links and their signs based on localized features; in fact they achieve higher accuracy for predicting links that have a higher embeddedness i.e. those that belong to a greater number of triangles. More recent work by DuBois *et al.* [43] is an extension of the link prediction problem studied in [85]. They develop a spring-embedding algorithm that they use in conjunction with their path-probability technique [42] to infer trust or distrust. They are able to achieve similar accuracy levels as in [85] when predicting undirected signed links but the use of a spring-embedding algorithm limits the application of their technique to directed networks.

All the above work are concerned about detecting the signs of current relationships that are not shown in the (incomplete) data set. Part of our study involves predicting *future* relationships. But the thrust of our work is to look at whether the opinions of our friends/foes have any correlation with the formation of new relationships.

### 5.2.3 Detecting Deceptive/Fake Reviews

Since consumers are increasingly relying on user-generated online reviews for making their purchase decisions [3], there has been increasing evidence of *deceptive opinion spam* - fictitious opinions that have been written to sound authentic. These deceptive opinions distort the perceived quality of the product and reduce the trustworthiness of online opinions in general. In recent years, a substantial body of work has been devoted to identifying and filtering out opinion spam, mostly using natural language processing and machine learning techniques. [74] identifies duplicate reviews, which are invariably fake in nature (since no two products/users would share/assign an identical review). Ott *et al.* in [108] designing a classifier using a given reference dataset as ground-truth. Feng *et al.* in [47] study the distributions of review rating scores to identify fake reviews. More recently, people start to look at structural information. Akoglu *et al.* in [6] exploit the connectivity structure

between review data and the reviewers. [107] studies the prevalence of deception spam across different online consumer review websites.

We remark that our work is along a different direction from all the previous work. We aim to identify and quantify the level of alignment between one's friends/foes opinions and the way he/she decides toward other users. Being able to quantify this correlation/alignment helps us build better models for recommending content as well as understand the interplay between the rating system and the social structure that supports it.

## 5.3 Dataset

We conduct our experiments on the community of users of the popular consumer review website Epinions. The website allows users to write reviews about products and services. It also allows users to rate reviews written by other users. In our dataset, each review can be rated on an integer scale of $[1-5]$ with a rating of $5$ signifying a very good review. Epinions also allows users to define their ***web of trust***, i.e. "reviewers whose reviews and ratings have been consistently found to be valuable" and their ***block list***, i.e. "reviewers whose reviews are found to be consistently offensive, inaccurate or not valuable". The *web of trust* and the corresponding *block list* for each user can be used to construct the directed, signed social network of trust and distrust.

| Statistic for Epinions | |
|---|---|
| Number of Persons | 131,828 |
| Number of Reviews | 1,197,816 |
| Number of Trust Edges | 717,667 |
| Number of Distrust Edges | 123,705 |
| Number of Ratings | 12,943,546 |
| Time Range | Jan-2001 to Aug-2003 |

Table 23: Overall Statistics for Epinions

Table 23 provides statistics for the Epinions dataset. The dataset contains

132k users who issued 840k statements of trust and distrust, wrote 1.2 million reviews and assigned 13 million ratings to the reviews. The dataset spans 2.5 years and it also contains timestamp information on trust/distrust edge formation as well as time of rating a review. About 85k users have received at least one statement of trust or distrust in the dataset. Trust (positive) edges dominate the social network, i.e. over 85% of the edges are trust ones.

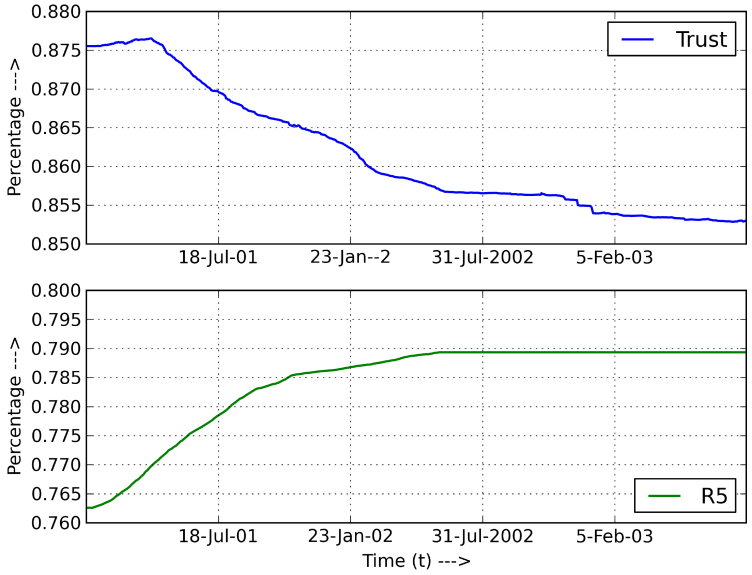



Figure 13: Progression of percentage of "trust" edges & percentage of R5 ratings over time.

Figure 13 shows the progression in the percentage of trust edges over time. As it is depicted, there is a drop in the overall percentage of trust edges indicating that users get more evolved into using distrust edges over time. In Epinions, when a user distrusts another user, reviews that are generated by the distrusted user will become hidden for him/her. Therefore, an increase in the usage of distrust edges would let users to see more of the reviews they like. This would increase the probability of assigning a rating of score 5 (R5) to a review by each user, which is shown in Figure 13 as well.

Figure 14 details various distributions that arise in the dataset. Summarizing the findings, we observe:

- A power-law degree distribution for both trust and distrust edges, i.e. which

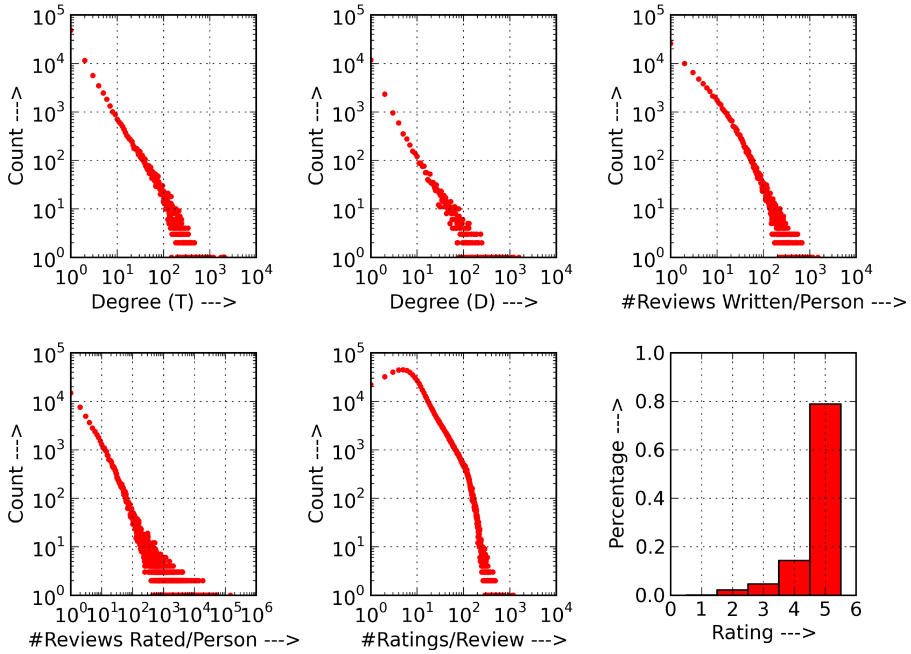



Figure 14: Distributions for Epinions Data. The plots (from left to right, top to bottom) indicate, (1) Degree Distribution of Trust Edges, (2) Degree Distribution of Distrust Edges, (3) Distribution of the Number of Reviews Written per Person, (4) Distribution of the Number of Reviews Rated per Person, (5) Distribution of the Number of Ratings per Reviews, and (6) Distribution of Ratings in the Dataset.

indicates that most users do not issue more than a handful of trust/distrust statements.

- A large number of users are passive users, i.e. users who write few or no reviews and rarely rate reviews.
- The bulk of content comes from a handful of users who take efforts on writing and rating reviews. We refer to these users as ***active users***.
- A rating of 5 is by far the most prevalent rating in the dataset. Over 78% of all ratings have a score of 5; on the other hand only 0.01% and 2.13% of ratings have scores of 1 and 2 respectively.

Since the dataset contains a handful of active users, we further conduct experiments to test the overlap between them. To do so, we measure the overlap among

top-$k$ active users by ranking them in two parts:

1. Rank active users based on the *number of trust statements received* and the *number of distrust statements received*.
2. Rank active users based on the *number of reviews written* and the *number of reviews rated*.

Measuring the overlap based on part (1), would help us detect any significant overlap between the most trusted active users and the most distrusted ones. While we would expect a non-zero overlap since leaders often polarize opinions, a large overlap would be detrimental to the objectivity of the system. Similarly, measuring the overlap based on part (2) would help us in observing any significant overlap between the top ***content generator*** (people who write reviews) active users and the top ***content consumer*** (people who rate reviews) active users. Again, a large overlap would be a disadvantage for the system since that would imply the existence of a closed group of content generators and consumers.

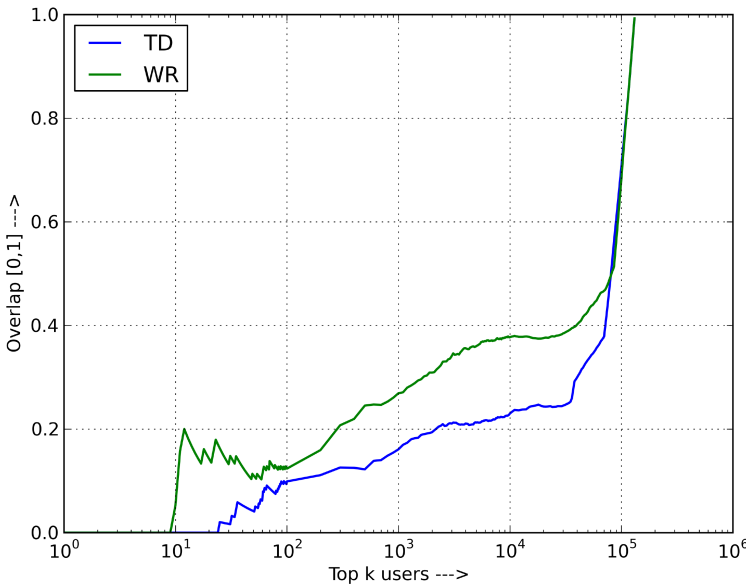



Figure 15: Overlap of top-$k$ users ranked by, (1)Trust & Distrust statements received (TD); (2) Number of reviews written & number of reviews rated (WR). Jaccard's Index [71] is used to compute overlap.

Figure 15 shows the overlap curves for both of the overlap measures. As it is depicted, there is a non-zero, small to moderate overlap for both of the measures for the top $1 - 10\%$ of the active users. Furthermore, although there is a small presence of ***controversial users*** (users who are liked and disliked in equal measure; refer to [94] for a comprehensive analysis of such users), a large majority of the top trusted active users are distinct from the top distrusted ones.

## 5.4 Relationship Formation

We investigate the correlation between the opinions of current friends and future relation formations under a simple scenario (see Figure 16). User $A$ is about to trust or distrust user $B$ at time $t$. At the moment, $A$ has $n$ friends/users, $F_1, F_2 \dots F_n$, whom he trusts. Among these $n$ friends/users, $b$ of them trust $B$ and $r$ of them distrust him. It is important to note that distrust relationships are hidden, i.e. $A$ does not know that "$r$" of his friends distrust $B$.

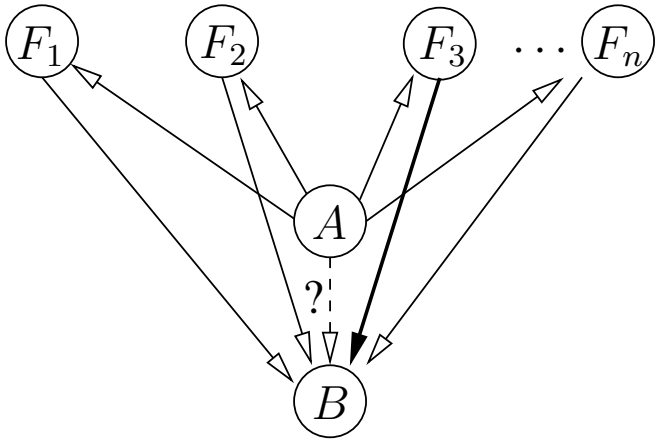



Figure 16: Relationship Formation Analysis Scenario. $A$ is about to decide on whether to trust or distrust $B$ at time $t$. $F_1, F_2 \dots F_n$ are friends of $A$ (i.e. people trusted by $A$), some of whom have already expressed trust or distrust of $B$. Thick links indicate distrust, others indicate trust.

We summarize here the results of this section:

1. First, we present raw observations based on the above scenario. Though informative, these observations should pass some processing to be statistically meaningful.
2. Then, we employ a random shuffling approach utilized in [86] to gauge whether the cases we observe are *over* or *under-represented* in the data relative to mere chance. Therefore, we compare our observed results with results achieved after randomly shuffling of the edge signs (trust/distrust). Besides over/under-representation, this approach allows us to determine statistical significance of our observed results.
3. Finally, our analysis should consider the fact that users in the data set exhibit a diverse spectrum of linking habits. Again, we use the approach put forward by [86]. We capture and gauge *trustfulness* and *trustworthiness* of users and employ them to get a better picture of the correlation between relationship formation and the opinions of friends.

### 5.4.1 Relationship Formation Scenario and Raw Observations

The following definition captures all the scenarios arising from Figure 16 in more detailed terminology. Depending on values of $b$ and $r$, there are four distinct possible cases.

**Definition 1.** *We categorize the cases as follows:*

1. $b > r$ *i.e. A's friends* ***collectively trust*** *user* $B$.
2. $b < r$ *i.e. A's friends* ***collectively distrust*** *user* $B$.
3. $0 < b = r$ *i.e. A's friends are* ***neutral*** *on whether to trust or distrust* $B$.
4. $0 = b = r$ *i.e. A's friends are* ***unopinionated*** *about user* $B$.

*We say that* $A$*'s friends are* ***opinionated*** *about his decision (trust/distrust) toward* $B$ *if either (1) or (2) happens. On the other hand, if either (3) or (4) happens, we say that* $A$*'s friends are* ***neutral****. Depending on the cases, we can figure out if* $A$ ***agrees/disagrees*** *with his friends in trusting* $B$.

| Category | Case | Friends Choose To | A's Decision | Count | Surprise | |
|---|---|---|---|---|---|---|
| Agreeing with friends (56.27%) | $b > r$ | Collectively Trust | Trust | 129,229 (49.25%) | 11.72 | 78.23 |
| | $r > b$ | Collectively Distrust | Distrust | 18,420 (7.02%) | 457.88 | |
| Disagreeing with friends (3.69%) | $b > r$ | Collectively Trust | Distrust | 6,656 (2.54%) | -101.98 | -110.97 |
| | $r > b$ | Collectively Distrust | Trust | 3,016 (1.15%) | -40.59 | |
| No role by friends (40.04%) | $0 < b = r$ | Conflict | Trust | 1,852 (0.71%) | -53.56 | -11.04 |
| | | | Distrust | 878 (0.33%) | -9.49 | |
| | $0 = b = r$ | Unopinionated | Trust | 76,636 (29.21%) | -36.95 | |
| | | | Distrust | 25,689 (9.79%) | 87.75 | |

Table 24: Evaluation of role played by friends in determining relationships relative to the random-shuffling model. Cases where $surprise > 0$ are marked in green and indicate over-representation; $surprise < 0$ cases are marked in yellow and indicate under-representation.

**Raw Observations** Our observations show that in majorities of cases there is an agreement between the opinions of a user's friends and his/her future relationships for majority of users. The detailed results are presented in Table 24. In more than $56\%$ of the cases, $A$'s decision to trust or distrust $B$ is aligned with his/her friends trust/distrust of $B$. Only in $3.69\%$ of the cases, $A$'s decision differs from his friend's trust/distrust of $B$[4].

In $40\%$ of the cases, $A$'s friends are unopinionated/neutral. The data show that a case of *neutral* (3) is very rare ($\approx 1\%$). We do not seek to test any correlation in the unopinionated/neutral cases.

Table 25 summarizes the data in Table 24 in terms of agreement rate. If $A$'s friends are *opinionated* about another user, then there is a strong correlation between $A$'s friends opinions and his/her decision ($93.85\%$); both the *collectively trust* and *collectively distrust* categories show very high agreement rates ($95.09\%$ and $85.93\%$ respectively). The *collectively distrust* case is specifically interesting because distrust links are *private* to other users.

[4]In our dataset, we observe that about $69\%$ of trust/distrust edges are observed on the first day of the crawl, i.e. they bear the same timestamp. Since there is no way of resolving the order in which these $69\%$ of the edges were formed, we treat the network induced by these edges as the snapshot of the network at time $t_0$. We conducted our relationship formation analysis for the remaining subset of edges ($31\%$) that are formed at time $t > t_0$.

| Category | Agreement Rate |
|---|---|
| Collectively Trust | **95.09%** |
| Collectively Distrust | **85.93%** |
| Opinionated | **93.85%** |

Table 25: The agreement rate of users in forming trust/distrust relationships in scenarios where their friends are opinionated.

### 5.4.2 Statistical Significance

We use a similar approach as in [86] to test the statistical significance of raw observations. In such an analysis we intend to gauge whether the cases we observe are *over* or *under-represented* in the data relative to mere chance. Therefore, we compare our observed results with results achieved after randomly shuffling of the edge signs (trust/distrust). In other words, if edge signs were produced at random (keeping the same proportion of trust and distrust edges), what would be the correlation between user's current relationships and his/her future ones?

**Definition 2** (From [86])**.** *We define the notion of* ***surprise****,* $\boldsymbol{s}$*, as the number of standard deviations by which the actual quantity differs from the expected number under the random-shuffling model. Formally, for a given scenario* $S$*, an actual quantity attached to the scenario* $Q$*, expected quantity* $E[Q]$ *and prior probability of the scenario* $p_0$ *under the random shuffling model, surprise* $s$ *is defined as follows:*

$$s = \frac{Q - E[Q]}{\sqrt{E[Q](1 - p_0)}} \tag{8}$$

A surprise value of $s > 0$ indicates over-representation whereas a value $s < 0$ indicates under-representation. A surprise value of $6$ yields a p-value of $\approx 10^{-8}$. In our experiments, we consistently encounter large surprise values, thereby making all our observations statistically significant.

Table 24 summarizes the results of the relationship formation analysis relative to the random shuffling model. One can clearly see over-representation in the "Agreeing with friends" scenario and under-representation in the other two scenarios. These results further support our findings:

1. There exists a strong correlation between a user's friends opinion and formation of his/her future relationships.
2. Such correlation exists even considering the distrust links which are the hidden links.

### 5.4.3 Validity of Observations with respect to Linking Habits

The data set include users with a diverse spectrum of linking habits, meaning that they might have great differences in the number of trustees/trusted links. We would like to see whether our observations (in previous subsections) hold even when we take these different habits into consideration. Again, we use the approach put forward by Leskovec *et al.* in [86].

**Definition 3** (From [86])**.** *We define* ***generative*** *and* ***receptive baselines*** *for a user as the fraction of positive links* created *and* received *by the user respectively. The baselines for a group of users are derived by summation of individual users' baselines in the group.*

The generative baseline for user $A$ is a measure of $A$'s *trustfulness* towards other users. Whereas, the receptive baseline for user $B$ is a measure of $B$'s *trustworthiness*.

We also need the notions of ***generative/receptive surprise*** values in order to check the validity of observations with respect to linking habits.

**Definition 4** (From [86])**.** *The* ***generative surprise*** *for a case is the number of standard deviations by which the actual number of positive $A \rightarrow B$ edges in the data differs above or below the expected number. The* ***receptive surprise*** *is defined along similar lines. If there was no correlation between friend of $A$'s opinions and his decision to trust or distrust $B$, we would expect surprise values of* $0$.

If $A$ made a decision in forming a relation with $B$ solely based on his linking habits (which is given by generative baseline), then *generative surprise* $= 0$. If $B$ is trusted/distrusted by people based on his receptive baseline, then *receptive surprise*$= 0$.

Table 26 provides the generative and receptive surprise values for all the cases of Definition 1. The observations show that $A$'s relationship with $B$ has a strong

| Case | Percentage of $A \rightarrow B$ Trust Edges | Generative Surprise | Receptive Surprise |
|---|---|---|---|
| Collectively Trust | 95.10% | **96.76** | **34.99** |
| Collectively Distrust | 14.07% | **-104.15** | **-56.31** |
| Conflict | 67.54% | -6.73 | -17.38 |
| Unopinionated | 74.89% | -26.87 | -27.94 |

Table 26: Generative & receptive surprise values for the 4 possible scenarios.

correlation with his friends opinions regarding $B$. This alignment might be due to homophily/heterophobia or social influence and our data set can not conclusively show which is the case. However, the existence of such alignments has consequences and applications for websites like Epinions.

- In the *collectively trust* case, both the generative and receptive surprise values are significantly positive. This means that $A$ exceeds generative baseline in trusting $B$ (which can be attributed to the fact that his friends have collectively trusted $B$ or just *homophily*). Also, $B$ exceeds it's receptive baseline in being trusted by $A$.
- In the *collectively distrust* case, both the generative and receptive surprise values are significantly negative. This means that $A$ falls behind his generative baseline in distrusting $B$ (which can be attributed to the fact that his friends have collectively distrusted $B$ or *heterophobia*). $B$ falls behind it's receptive baseline in being distrusted by $A$.

## 5.5 Ratings and Friend of Friend Dynamics

We extend the analysis to investigate the existence of any correlation between the opinions of the web of trust of a user and the opinions he/she expresses towards other users. As mentioned before, users of Epinions rate reviews of other users. We seek to determine if there is any correlation between the friendship dynamics and the ratings of a user. We will refer to the problem as analyzing ***friend-of-friend***

***(FoF) dynamics***, though it will also include the remaining three cases, namely, ***friend-of-enemy (FoE)***, ***enemy-of-friend (EoF)*** and ***enemy-of-enemy (EoE)***.

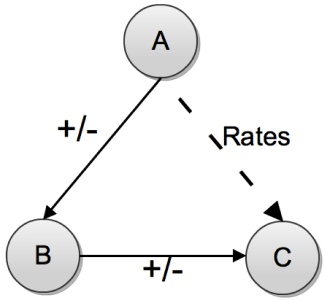



Figure 17: Analysis Scenario. User $A$ rates a review written by $C$ such that, (1) $A$ has not expressed any trust/distrust of $C$ at the time of rating, (2) at least one of $A$'s friends/foes have expressed trust/distrust of $C$.

**Definition 5** (***friend-of-friend (FoF) dynamics***)**.** *We define the problem using Figure 17. Consider three individuals $A$, $B$ and $C$. $A$ has expressed trust/distrust of $B$ at time $t_1$ and $B$ has expressed trust/distrust of $C$ at time $t_2$ (without loss of generality we can assume that $t_1 < t_2$). $A$ chooses to rate a review written by $C$ at time $t_3$ ($t_1 < t_2 < t_3$).*

*We would like to quantify the correlation between $A$'s rating and the opinion of $B$ regarding $C$. Notice that $A$ has **NOT** expressed any opinion of trust or distrust about $C$ at time $t_3$.*

The existence of a correlation might be used to improve recommendation systems (i.e. show content that would be useful to the user)[5].

There are four possible scenarios based on Figure 17. These four scenarios are depicted in Figure 18: FoF, EoF, FoE and EoE dynamics. In our dataset, $55\%$ ($\approx 7.2$ million) of all ratings fall under at least one of these four scenarios (a rating can fall under multiple scenarios depending on $A$'s friends/foes being friends or foes with $C$). Table 27 provides the raw observations for each scenario. As illustrated in the

[5]It is important to point out that this analysis differs from the one carried out by Leskovec *et al.* in [86]. They investigate signed triads in online social networks and analyze the over or under-representation of various triadic cases by applying theories from social psychology. The reviews and subsequent ratings by users do not appear in their analysis.

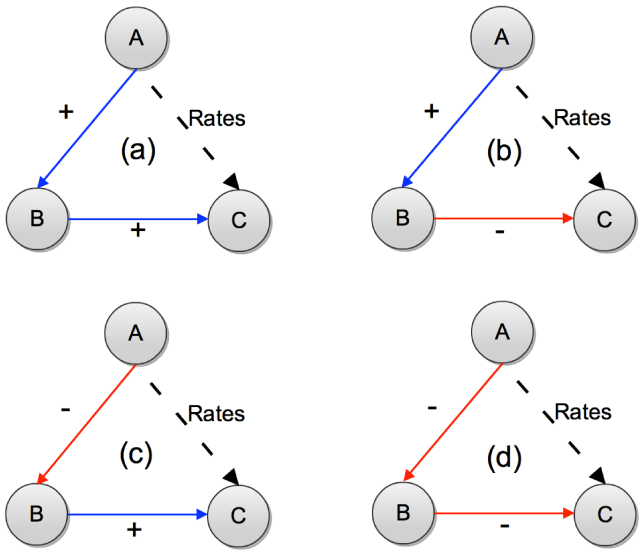



Figure 18: Scenarios in friend-of-friend dynamics. In scenario (a), $A$ rates a review written by $C$ such that $C$ is a friend-of-friend of $A$. Also $A$ hasn't expressed any trust or distrust in $C$ at the time of the rating. The remaining scenarios deal with enemy-of-friend, friend-of-enemy and enemy-of-enemy cases respectively. Trust edges are colored blue; distrust edges are colored red.

table, the FoF scenario dominates (over $92\%$) the other three scenarios, while the EoE (under $1\%$) scenario being the most rare scenario.

Again we investigate the statistical significance of raw observations by using random shuffling model [86]. We also consider the different rating/rate-ability habits of users by incorporating the generative/receptive baselines approach of [86].

1. We shuffle the edge signs and compute the surprise values for each of the four scenarios. The fourth row in Table 27 illustrates the observations. These values indicate that the FoF scenario is over-represented in the dataset relative to chance, whereas the other three scenarios (EoF, FoE, EoE) are under-represented.

2. We keep the edge signs intact while shuffling the ratings around (keeping the same distribution of the ratings shown in figure 14 ). The random shuffling of the ratings allows us to study the over or under-represenation of each rating in each scenario relative to chance (i.e. rating being assigned at random). Table

27 provides the raw counts, percentages and surprise values associated with ratings (as numbers in $\{1, 2, 3, 4, 5\}$) under each scenario.

3. To get a measure of the rating habits of a user, we extend the definitions of generative and receptive baselines defined in definition 3. Each user is associated with five ***generative/receptive rating baselines*** based on five categories of ratings. A ***generative rating baseline*** of $0.80$ associated with a user $A$ for rating of score $5$ would imply that the user $A$ assigns a rating of score $5$ in $80\%$ of times. Similarly, a ***receptive rating baseline*** of $0.20$ associated with a user $C$ for rating of score $4$ would suggest that user $C$ receives a rating of score $4$ for $20\%$ of his reviews. The generative rating baselines for user $A$ is a measure of $A$'s opinions towards reviews written by other users, whereas the receptive rating baselines for user $C$ are a measure of other users's opinions of reviews written by $C$. We can now compute the generative rating surprise and receptive rating surprise values associated with each rating under each scenario. Table 28 provides the ***generative rating surprise*** and ***receptive rating surprise*** values associated with each rating under each scenario.

### 5.5.1 Findings

We find the following by computing the under/over representation of each rating in each scenario (please refer to Tables 27 & 28 for the surprise values): i) Low ratings (ratings of scores $1$ and $2$) are over-represented in all scenarios except the FoF one; ii) Rating of score $4$ is under-represented across all scenarios; iii) Rating of score $5$ (the highest and most frequent rating) is over-represented in all scenarios except the EoF one (in which it is under-represented). Based on our findings, we observe the following trends:

1. We see a clear trend of alignment between ratings and opinions of friends/foes in the FoF and EoF scenarios:

   **FoF**: We see a shift towards higher ratings (rating of score $5$ is over-represented and ratings of scores $1$ and $2$ are under-represented in this scenario). These results suggest that user $A$ is more likely to assign higher ratings to user $C$'s reviews when $C$ happens to be a friend of a friend of $A$. The

generative and receptive surprise values also indicate such a correlation.

**EoF**: We see a trend towards assigning lower ratings (ratings of scores 1 and 2 are over-represented whereas all other ratings are under-represented). This indicates that $A$ would be more likely to assign lower ratings to $C$'s reviews when $C$ happens to be an enemy of a friend of $A$. Again, the generative and receptive surprise values support such a correlation.

| Scenario | | $A \xrightarrow{T} B \xrightarrow{T} C$ | $A \xrightarrow{T} B \xrightarrow{D} C$ | $A \xrightarrow{D} B \xrightarrow{T} C$ | $A \xrightarrow{D} B \xrightarrow{D} C$ |
|---|---|---|---|---|---|
| Dynamics | | FoF | EoF | FoE | EoE |
| Count (c) | | 77,801,592 | 2,225,299 | 3,069,957 | 746,361 |
| Percentage (p) | | 92.79% | 2.65% | 3.66% | 0.89% |
| Surprise (s) | | 4163.27 | -2760.75 | -2466.35 | -826.46 |
| R1 | c | 1585 | 1411 | 1053 | 134 |
| | p | 0.002% | 0.06% | 0.03% | 0.02% |
| | $s$ | -87.42 | 67.24 | 30.72 | 3.05 |
| R2 | c | 109,193 | 216,251 | 75,253 | 28,927 |
| | p | 0.14% | 9.72% | 2.45% | 3.88% |
| | $s$ | -1214.14 | 782.66 | 36.66 | 103.25 |
| R3 | c | 6,242,221 | 88,495 | 53,398 | 14,742 |
| | p | 8.02% | 3.98% | 1.74% | 1.98% |
| | $s$ | 1440.39 | -45.06 | -244.93 | -111.09 |
| R4 | c | 8,825,021 | 240,785 | 247,666 | 55,783 |
| | p | 11.34% | 10.82% | 8.07% | 7.47% |
| | $s$ | -755.62 | -147.63 | -309.12 | -165.92 |
| R5 | c | 62,623,572 | 1,678,357 | 2,692,587 | 646,775 |
| | p | 80.49% | 75.42% | 87.71% | 86.66% |
| | $s$ | 341.32 | -129.48 | 377.85 | 162.89 |

Table 27: Random-Shuffling Model Analysis. R[1-5] refers to the five categories of possible ratings. Surprise values indicate over/under representation relative to chance under the random-shuffling model [86]. Colors indicate whether the surprise values are greater than (green) or less than (yellow) 0. We see a shift towards assigning higher ratings (score 5) in the FoF scenario and a shift towards assigning lower ratings (scores 1 & 2) in the EoF scenario.

2. In the remaining two scenarios of FoE and EoE, we see a divided picture. Both low ratings (scores of 1 and 2) and high rating (score of 5) are over-represented whereas the ratings in the middle (scores of 3 and 4) are under-represented. For these two scenarios, we are not able to conclusively show signs of correlation from this analysis.

| Scenario Dynamics | | $A \xrightarrow{T} B \xrightarrow{T} C$ FoF | $A \xrightarrow{T} B \xrightarrow{D} C$ EoF | $A \xrightarrow{D} B \xrightarrow{T} C$ FoE | $A \xrightarrow{D} B \xrightarrow{D} C$ EoE |
|---|---|---|---|---|---|
| R1 | $s_g$ | -43.77 | 66.03 | -8.36 | -1.43 |
| | $s_r$ | -10.91 | 19.80 | 57.13 | 2.76 |
| R2 | $s_g$ | -627.54 | 789.36 | -108.83 | 26.54 |
| | $s_r$ | -527.77 | 89.58 | 206.16 | 4.03 |
| R3 | $s_g$ | -360.72 | 2.01 | -304.42 | -124.23 |
| | $s_r$ | -181.17 | 10.16 | 65.53 | 5.89 |
| R4 | $s_g$ | -847.21 | -115.23 | -381.94 | -190.69 |
| | $s_r$ | -370.22 | -3.57 | 81.06 | -4.27 |
| R5 | $s_g$ | 1065.09 | -189.93 | 531.88 | 214.75 |
| | $s_r$ | 519.91 | -61.03 | -173.88 | -1.36 |

Table 28: Rating Habits Analysis. R[1-5] refers to the five categories of possible ratings. Surprise values indicate over/under representation relative to the rating habits of the users. Quantities $s_g$ & $s_r$ indicate the generative/ receptive rating surprise values respectively. Colors indicate whether the surprise values are greater than (green) or less than (yellow) 0. We again observe a shift towards assigning higher ratings (score 5) in the FoF scenario and a shift towards assigning lower ratings (scores 1, 2 & 3) in the EoF scenario.

## 5.6 Building a Predictor Model

In order to further substantiate our findings, we use FoF Dynamics to build a predictor model and apply it to predict the rating assigned by a user to another user's review. Such a model could also serve as a recommender system by recommending content that is ***likely*** to be rated well by a particular user. Table 29 details the

features that we use to build such a model. The features can be classified into three distinct categories: (1) ***Trust Features*** that measure the *trustfulness* and *trustworthiness* of the users involved; (2) ***Rating Features*** which capture the rating habits of the users; and (3) ***FoF Dynamics Features*** which intends to utilize the correlation between a user's rating and his friends opinions.

Since the dataset did not contain the actual reviews, we do not use any content features that could analyze the actual content of the reviews.

### 5.6.1 Correlation Coefficients of Features

First, we calculate the Spearman's rank correlation coefficient between the features and the actual rating. The correlation coefficient allows us to determine the direction of association between our features and the actual rating. A value of $0$ for a feature would indicate that there is no relation between that feature and the actual rating. On the other hand, a positive correlation coefficient would indicate that an increase in the feature value is accompanied by an increase in the actual rating (and vice-versa). The magnitude of the correlation indicates the strength of this association[6].

Table 29 shows the correlation coefficients for the different features. The findings are along expected lines. They are as follows

1. The *trustworthiness* of user C is positively correlated with the actual rating received. Thus a trustworthy user is likely to receive higher ratings for his reviews.
2. The rating tendency of user A is positively correlated with the actual rating i.e. a user is likely to rate current content based on his past rating trends. Similarly, the average rating received by C for his past reviews is a good indicator of the rating that he is likely to receive for his future reviews.
3. Along with the trust & rating features, we observe significant correlations for the FoF & EoF features. These results are in line with our earlier findings in section 5.5.

[6]We choose Spearman's coefficient [128] over Pearson's coefficient due to the skewed nature of the ratings (refer to figure 14 for the distribution of ratings) as well as to guard against outliers.

| Feature Class | Feature | Meaning | Correlation Coefficient |
|---|---|---|---|
| Trust | A's Generative Baseline | The generative baseline for a user is a measure of his trustfulness towards other users. The receptive baseline for a user is a measure of his trustworthiness in other users eyes. | -0.0036 |
| | C's Generative Baseline | | **-0.1840** |
| | A's Receptive Baseline | | 0.0236 |
| | C's Receptive Baseline | | **0.0771** |
| Rating | Avg. Rating given by A | The average rating given by a user captures his rating tendencies. The average rating received by the reviews written by a user captures the usefulness of the content generated by the user. | **0.2555** |
| | Avg. Rating given by C | | 0.0431 |
| | Avg. Rating received by A's reviews | | **0.1286** |
| | Avg. Rating received by C's reviews | | **0.3606** |
| FoF Dynamics | Number of FoF Paths | These features capture the number of FoF, EoF, FoE & EoE paths between two users. The features are useful in providing context in the absence of a direct link. | **0.1112** |
| | Number of EoF Paths | | **-0.0918** |
| | Number of FoE Paths | | 0.0105 |
| | Number of EoE Paths | | -0.0001 |

Table 29: Spearman's Rank Correlation coefficient between feature values and actual ratings. Correlation coefficients with significant amplitude ($> 0.05$) are highlighted in colors, red for negative correlation, and blue for positive correlation. Here user A is assigning a rating to a review written by user C.

### 5.6.2 Predicting Ratings

Having described the utilized features, we now describe the techniques used to build the predictor model and also list the performance of these techniques. First, instead of directly using the actual rating as the class label, we define the following three classes of ratings: (1) ***High ratings*** which include ratings of scores 4 & 5; (2) ***Medium ratings*** which includes ratings of score 3; and (3) ***Low ratings*** which include ratings of scores 1 & 2. Given the feature samples, our predictors will attempt to predict these class labels.

Due to the skewed nature of the ratings towards high ratings, if we randomly select training samples, the predictive model will overfit to the *high ratings* class, and may fail to predict other classes. To avoid overfitting, we take a balanced sampling approach and generate a training set containing roughly an equal amount of samples from each class. In our experiments, we select $25,000$ samples from each of the three classes.

In addition to the correlation coefficient analysis, we also determine the relative importance of the features by computing the mutual information between a feature and the class label. Table 30 lists the results of this analysis. Qualitatively,

| Class | Feature | Information Gain |
|---|---|---|
| Trust | A's Generative Baseline | 0.1595 |
| | C's Generative Baseline | 0.2291 |
| | A's Receptive Baseline | 0.1943 |
| | C's Receptive Baseline | **0.4496** |
| Rating | Avg. Rating given by A | 0.3316 |
| | Avg. Rating given by C | **0.3776** |
| | Avg. Rating received by A's reviews | **0.2453** |
| | Avg. Rating received by C's reviews | **0.5362** |
| FoF Dynamics | Number of FoF Paths | **0.3813** |
| | Number of EoF Paths | 0.1894 |
| | Number of FoE Paths | 0.0119 |
| | Number of EoE Paths | 0.0198 |

Table 30: Mutual Information between features and the rating classes (high, medium & low). Top-5 features are highlighted in blue.

tables 29 & 30 are in rough agreement.

Finally, to predict the rating classes, we experimented with a number of classification techniques using Weka [58]. Overall, tree-based classification techniques have worked well in our case. Figure 19 reports the prediction accuracy for the different techniques. The overall prediction accuracy is around $76\%$. The ROC area (AUC) is an impressive $0.91$. This is significantly better than random guessing (with $33\%$ accuracy). This result indicates that our feature set (that includes the FoF Dynamics Features) is predictive of the rating class, and the classifier can be used to predict ratings reliably. In fact, when ranked based on information gain, the FoF feature is among the top-3 features.

We also formulated rating prediction as a regression problem, meaning that we constructed a mapping from the feature set to the actual rating value. Regression performance is measured using common metrics such as mean absolute error (MAE) and mean squared error (MSE). The correlation coefficient between the predicted value and the true target value can also be considered as a metric of accuracy, with high value indicating good performance. In Epinions dataset, we achieve a

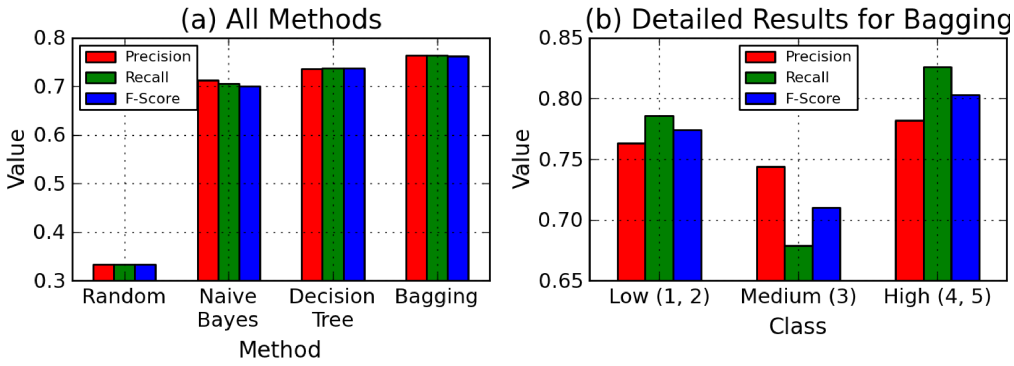



Figure 19: Prediction results using different classification techniques. The figures plot precision, recall & f-measure for the different classifiers (left) and the detailed results for Bagging across the three rating classes (right). Bagging [24] is an ensemble method which improves the classification accuracy through sampling and model averaging. Bagging provides an accuracy in excess of $76\%$ with an ROC area (AUC) of $0.91$.

MAE of $0.5541$, a MSE of $0.7893$ and a correlation coefficient of $0.7646$. This indicates that the regression accuracy is also quite good.

We conclude that the feature set combining individual trust, rating & social dynamics features can support rating prediction, and that our approach leads to good prediction and classification accuracy.

## 5.7 Conclusion

In this paper, we looked at two distinct problems of alignment/correlations in the interplay of eCommerce and social networks. Firstly, we looked at the correlation between one's current friends and his/her future relationships. Our findings suggest that users are more likely to decide future relationships in alignment to the opinions of their current friends. The interesting observation was that this alignment exists not only in choosing future friends, but also in choosing future foes.

In our second analysis, we studied the alignment between the FoF dynamics and a user's rating of others content. We concluded that users are more likely to rate the content of a third person in alignment with the opinions of his/her friend

regarding the third person. Our findings also show that the opinions of foes have little or no correlation with ratings of a user.

We also built a model that can predict the rating assigned by a user to a review. This model is able to achieve a good accuracy of $76.34\%$ and an AUC of $0.91$. Our predictor could also be used to recommend content to Epinions users.

Studying correlation and correctly measuring is very important in design and analysis of rating systems. We have showed that such a study can be utilized to improve the user experience in similar systems or websites. From the sociology point of view, both of the above alignments can be explained by social influence or homophily/heterophobia. However, we should mention that our analysis is not conclusive in favor of either explanation.

# Chapter 6

# Noisy Graph Matching using Laplacian Based Descriptors

## 6.1 Introduction

In this chapter we consider the problem of matching two anonymous networks that share a majority of their nodes and their edge sets differ, albeit only slightly. This problem appears in a variety of social network analysis settings. Various snapshots of communication traces among terrorists may be used to figure out who is who even when multiple modes of communication (e.g. cell phones, email) are used or individual terrorists employ multiple user accounts or access points. An individual can be active in multiple social contexts and may replicate the same social interaction patterns across different platforms (such as Facebook and Flicker). Matching the two graphs may lead to unifying information obtained in each network to put together a more complete picture of their social interactions. In knowledge based networks (such as Wikipedia), the same concepts can be expressed in multiple networks using different languages. Matching the network structures can be useful for machine translation. In another example, one may want to verify that one network cannot be de-anonymized through matching with a second non-anonymous network embedding the same set of nodes.

In social networks, people with different roles or status levels often have different kinds of structural positions. In a co-authorship network, for example, scholars

who are field leaders or "big names" in the community are often well connected to other members with high influence within the community, compared to newcomers. In the context of viral diffusion, nodes who are at a higher risk of being infected are often more active. At the graph topology level, some nodes are close to the "core" of the network, while some others are loosely connected and remain at the network "periphery". It has been a common practice to use a node's degree to get a hint of its social status and its level of 'connectivity' within the network. For example, nodes with higher degree are considered more contagious when it comes to the study of epidemics [46]. But node degree is a local property and the position of a node in the network clearly cannot be described only by its local neighborhood. Some other features, such as betweenness centrality [139] and pagerank, have been developed with information on the global network topology. But it is unclear how descriptive they are in differentiating different nodes. In this chapter we develop a multi-resolution, compact signature that encodes not only the information on a node's immediate neighborhood but also its position in an increasingly larger neighborhood and the global network topology.

The difficulty of the graph matching problem considered depends on the degree of differentiation in structural positions between nodes. An important concept for present purposes is therefore that of "regular equivalence" [103]. Two nodes who are regularly equivalent do not necessarily share the same neighbors, but have neighbors *who are themselves similar*. The problem of measuring regular equivalence has been studied in a number of papers [18, 72, 84, 139, 145]. In the algebraic solutions, the similarity between two nodes $i, j$ boils down to "a weighted count of all paths between $i$ and $j$ with paths of length $r$ getting weight $\alpha^r$". We remark that this definition of similarity only applies on a *pair* of vertices in a *single* graph. It is unclear how to compare two nodes in *different* but highly similar graphs. Nevertheless, the more nodes in a graph that are regularly equivalent the harder it will be to match two instantiations or perturbations of that graph as we can expect these nodes will score identically on *any* measure of relative structural positions.

### 6.1.1 Related Work

The problem we study here is related to the classic *graph isomorphism problem* [82], where the goal is to decide if there is a 1-1 correspondence between the nodes that preserves all edges. Amazingly, this is a well-studied problem in the algorithms community whose complexity nevertheless has remained undetermined, even though almost all generalizations (such as the largest common subgraph problem) are known to be NP-complete. Efficient algorithms are available only for special cases. Examples include all kinds of graphs that admit a canonical form, including trees and planar graphs, and the graphs where unique labels can be computed on vertices or other aspects of the graph, in which case matching the labels to each other suffices to identify an isomorphism. Examples of the second class are graphs in which all vertices have different degree, or the eigenvalues of the graph Laplacian are unique [133]. After identifying these special cases, research has focused on extending the algorithms in one of two ways: by producing an algorithm whose run time is exponentially bounded by a specific property of a graph that is small for most graphs, or by producing an algorithm that is only guaranteed to give correct answers for the special case but can still provide useful answers in the general case. An example of the former is an algorithm for matching graphs in time exponential in the genus of the graph [97], which is thus polynomial on graphs of bounded genus. An example of the latter are spectral techniques that improve the quality of the matching when there are repeated or similar eigenvalues by, for example, solving an LP relaxation of an optimization formulation of the problem [7,153]. In fact, these algorithms tend to perform well on all except a few classes of graphs, one of which is the class of regular graphs, or graphs where all vertices have the same degree. Intuitively, this is a difficult case because all of the vertices are locally indistinguishable from each other. In our case we are interested in "noisy" graphs coming from real-world data, so results in this area are not directly applicable.

The problem of comparing graphs has also been studied in the machine learning community, but with the different goal of obtaining a similarity metric between graphs, often referred to as a *graph kernel* [50,78,119,126,127]. Our setting, however, differs from the graph kernel setting in that we are much more interested in explicit informative correspondences between graphs than a single score of how similar they are.

A couple of practical algorithms for matching noisy social networks have been proposed in [147, 148], in which a subset of nodes in different networks are preliminarily revealed and identified. Given some nodes that are already matched, the problem of matching two networks becomes much easier since one can 'position' each unmatched node relative to the set of the given matched, anchor nodes. In our setting, we have no knowledge of any node matches initially and the challenge is to extract the position of a node purely from its relationship to other nodes in the graph topology. We remark though that we can also exploit the information of a subset of identified nodes in two graphs.

Lastly, we note that our graph matching problem is related to a number of recent studies on de-anonymization of social networks [11, 102]. It is a standard practice to block out user identifying information such as name and IP addresses before a social network data set is made public. Our scheme can be considered as a de-anonymizing technique for an anonymous network, when a similar, non-anonymous network is provided.

### 6.1.2 Our Contribution

In this study we formulate a generic family of graph signatures, called the *Laplacian family signatures (LFS)*, that captures the multi-scale topological information centered at each node. This family includes the kernel signature (HKS) [130], wave kernel signature (WKS) [10], wavelet signatures (WS) [59] and others. In particular, the heat kernel signature is the diagonal of the heat kernel. The heat kernel signature for a node $i$ is a curve parameterized by $t$ and describes the amount of heat returning to the heat source $i$, after time $t$. It can also be understood as the probability of a random walk coming back to the starting point $i$ after time $t$. Therefore it is naturally multi-resolutional and can be considered as a way of defining the position of $i$ with respect to the entire network. Wave kernel signature and wavelet signatures have different physical motivations but when it comes to the formula they are very similar to HKS. All of them use a weighted sum over squared eigenvectors. Therefore we formulate the entire family of signatures as weighted sums of squared signatures and use them for matching noisy graphs, based on the intuition that the same node in different graphs have similar signatures.

This family of signatures were originally proposed for points on smooth manifold. In the smooth setting, both HKS and WKS were shown to contain *all* of the information about the intrinsic geometry of the shape and hence *fully* characterize shapes up to isometry [59, 130]. In a discrete graph, it is also known that local random walks (essentially heat diffusion) can derive information about global properties of the graph such as the graph genus [16]. In this study, it is the first time that these signatures are examined in a discrete graph setting and applied to matching noisy graphs. Obviously, a desirable property of signatures for graph matching purpose is the stability under difference noise models. Here, we consider noises on edges weights, and addition/removal of weak nodes[1]. Under some mild conditions, we can show analytically that LFS is indeed a stable descriptor and therefore establish the theoretical foundation of using LFS for matching two noisy graphs.

Given two unlabeled graphs $G_1, G_2$ with similar graph topology, we would like to find a mapping of the vertices such that the patterns of connectivity in the two graphs roughly match. We first compute the LFS vectors for the nodes in both graphs and then map nodes in $G_1$ to the nodes in $G_2$ with similar LFS. We evaluate the idea of matching noisy graphs by using both artificially generated model graphs (Erdos-Rényi model, Barabási-Albert Preferential Attachment model, Watts-Stragotz model, Kleinberg's hierarchical small world model, etc) and real-world data sets on co-citation networks and co-authorship networks. Our experiments show that within the family, matching performance may vary, our comparison, however, results in that the best descriptor in the family always outperforms other alternative graph matching algorithms, such as the Umeyama algorithm [133], the Rank algorithm [122], and the QCV (quadratic convex relaxation) algorithm and the PATH algorithm [150].

## 6.2 Laplacian Family Signatures

In this section, we will formally introduce the definition of Laplacian family signatures (LFS) and the continuity property under different noise models.

[1]Weak nodes means a set of nodes that are loosely connected to the rest of the graph, namely the connection between the set of nodes and the rest of the nodes are small compared with the connections within the rest of the nodes.

### 6.2.1 Preliminary on Graph Laplacian

Given an undirected graph $G = (V, E)$, where $V$ is the set of vertices, and $E \subseteq V \times V$ is the set of edges. Let $w$ be the weights on edges, i.e. $w : E \mapsto \mathbb{R}^+$. The graph Laplacian is defined as $\mathcal{L} = D - A$, where $A$ is the graph weight matrix with

$$A_{ij} = \begin{cases} w(i,j) & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$$

and $D$ is a diagonal matrix of total incident weights, i.e. $D_{ii} = \sum_j A_{ij}$. For unweighted graph, $A$ becomes the adjacency matrix and $D$ is the diagonal degree matrix. Here, we consider only simple graphs, i.e. graphs without self-loops and multiple edges.

Here, we will mention several properties of $\mathcal{L}$ that would be useful in our analysis, interested reader could refer to [17] for a list of other properties.

**Proposition 1. (Properties of $\mathcal{L}$)**

*(i)* *$\mathcal{L}$ is symmetric and positive semi-definite.*

*(ii)* *The smallest eigenvalue of $\mathcal{L}$ is $0$, and the multiplicity corresponds to the number of connected components of $G$. If $G$ is simply connected, $\phi_1 = \frac{1}{\sqrt{|V|}}\mathbb{1}$.*

*(iii)* *$L$ has non-negative, real-valued eigenvalues $0 = \lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{|V|}$.*

Graph Laplacian $\mathcal{L}$ has an important role in spectral graph theory. Similar to classical Fourier transform, $\mathcal{L}$ can be used to define the graph Fourier transform [59], where the eigenvalues $\{\lambda_i\}$ are discrete Fourier frequencies and the eigenvectors $\{\phi_i\}$ form the graph Fourier basis. Analogously, small eigepairs are low frequency components of the graph that capture more the global property, while large eigenpairs, on the contrary, focus more on the local property of the graph.

### 6.2.2 HKS, WKS, WS

The *heat kernel signature (HKS)* of $G$ [130] is defined as the diagonal of the heat kernel. HKS for node $l \in G$ is

$$s_l^{\text{HKS}}(t) = \sum_k \exp(-t\lambda_k)\phi_{lk}^2. \tag{9}$$

The *wave kernel signature (WKS)* of $G$ [10] for node $l \in G$ is defined as

$$s_l^{\text{WKS}}(t) = \sum_k \exp(-\frac{(t - \log \lambda_k)^2}{2\sigma^2})\phi_{lk}^2. \tag{10}$$

where $\sigma$ is the variance. Both HKS and WKS are proved to work nicely for smooth manifolds. To visualize how distinctive they can be on well-structured graphs, we build a balanced binary tree and plot the HKS and WKS for each level in Figure 20 which illustrates the describing power of both signatures.

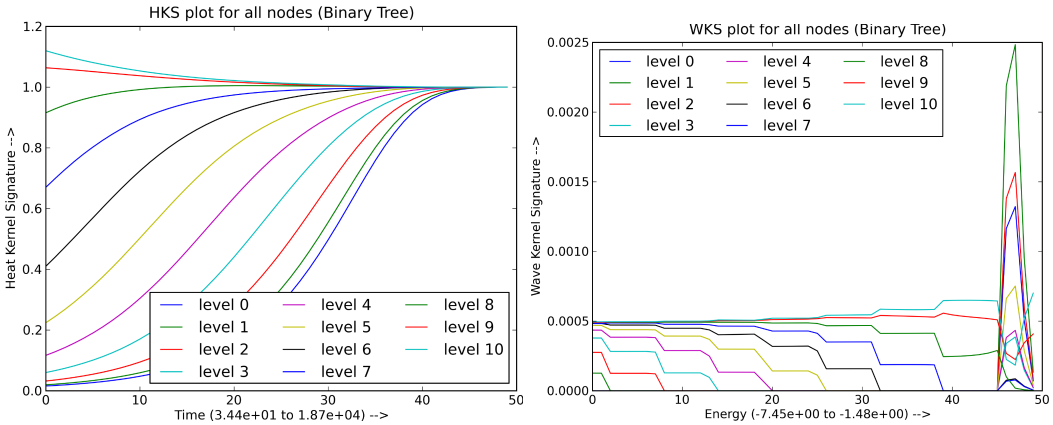



Figure 20: Heat and Wave Kernel Signatures on a Binary Tree.

Wavelet on graph [59] is defined from a kernel $g(\cdot)$ on $\mathcal{L}$ that satisfies admissibility condition $\int_0^\infty \frac{g^2(x)}{x}dx = C_g < \infty$ with $g(0) = 0$. The *wavelet signature (WS)* can thus be similarly defined as

$$s_l^{\text{GLS}}(t) = \sum_k g(t\lambda_k)\phi_{lk}^2 \tag{11}$$

### 6.2.3 Laplacian Family Signatures

Noticing the similarity in the form of HKS, WKS, and WS, all can be seen as a weighted sum over squared eigenvectors. We could therefore formulate the generic form of LFS as

$$s_l(t) = \sum_k h(t; \lambda_k)\phi_{lk}^2 \tag{12}$$

where $h(t; \lambda_k)$ is the construction kernel.

To see the limit of the description power of $h(t; \lambda)$, we have the following theorem.

**Theorem 1** (Order Theorem)**.** *If d is the diameter of the graph, the maximum informative order of $h(t;\lambda)$ on $\lambda$ is 2d.*

In this study, we consider specifically several kind of kernel functions, some of which are highly related to the HKS and WKS.

1. Gamma kernel $\Gamma(k,\theta) \propto x^{k-1}\exp\left(-\frac{x}{\theta}\right)$. Note HKS is a special case of Gamma kernel with $k=\theta=1$.
2. Gaussian kernel $G(\mu,\sigma) \propto \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$. WKS is a special case of Gaussian kernel with $x=t,\ \mu=\log\lambda_j$.
3. Laplacian kernel $L(\mu,\sigma) \propto \exp\left(-\frac{|x-\mu|}{\sigma}\right)$.
4. Generalized Rayleigh kernel $R(k,\theta) \propto x^{k-1}\exp\left(-\frac{x^2}{2\theta^2}\right)$.
5. Generalized Student's t kernel $t(k,\theta) \propto \left(1+\frac{x^2}{\theta}\right)^{-\frac{k+1}{2}}$.
6. Generalized Inverse Chi square ($\chi^{-2}$) kernel $\chi^{-2}(k,\theta) \propto x^{-\frac{k}{2}-1}\exp\left(-\frac{\theta}{2x}\right)$.

### 6.2.4 Distance of LFS

Distance between two LFS can be naturally defined as the $L_2$ distance

$$d_{L_2}(s_i,s_j)^2 = |s_i(t)-s_j(t)|_2 = \int_0^\infty |s_i(\tau)-s_j(\tau)|^2 d\tau.$$

For the discrete calculation, each LFS is sampled at time instances $t_{\text{min}} = t_0 < t_1 < \cdots < t_s = t_{\text{max}}$, where $t_i$'s are discrete samples[2]. Then the $L_2$ distance between the LFS at two nodes could be well-approximated by

$$d_2(s_i,s_j)^2 = \sum_{k=0}^{L} |s_i(t_k)-s_j(t_k)|^2.$$

$L_2$ distance reflects more the absolute difference upon each time slice $t$, while [10] defined a distance function that concentrates on the relative difference on each $t$, and we call it WKS distance

$$d_w(s_i,s_j)^2 = \sum_{k=0}^{L} \frac{|s_i(t_k)-s_j(t_k)|}{s_i(t_k)+s_j(t_k)}.$$

[2]In the implementation, $t_i$'s are equally distributed in log-scale, except WS in linear-scale.

### 6.2.5 Continuity of LFS

Here we discuss the continuity of LFS on a weighted graph under small perturbations. We consider three noise model, namely edge noise, addition/removal of weak nodes. To have the continuity property, the construction kernel $h(t; \lambda_k)$ must satisfy some mild continuity condition. LFS continuity then could be seen as an immediate consequence of the continuity of graph laplacians under these conditions.

**Theorem 2** (Continuity of Laplacian on Weighted Graph)**.** *Let $\mathcal{L}_1$, $\mathcal{L}_2$ be the graph Laplacian for $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ respectively. Let eigenvalues of $\mathcal{L}_i$ be $\{\lambda_k^{(i)}\}_{k=1}^n$ in the order of increasing value with multiplicity, for $i = 1, 2$, and the corresponding eigenvectors are $\{v_k^{(i)}\}_{k=1}^n$. If*

1. *(well separated eigenvalues) $\exists\ \delta > 0$, such that $\lambda_{i+1} - \lambda_i > \delta,\ \forall i$,*
2. *(bounded changes) $|\mathcal{L}_1 - \mathcal{L}_2|_F < \epsilon$,*

*then for each $k$,*

1. *$|\lambda_k^1 - \lambda_k^2| < \epsilon$,*
2. *if $v_k^2 = v_k^1 + \Delta v_k$, then $||\Delta v_k||_2 < C(\delta)\epsilon$ for $C(\delta)$ independent of $\epsilon$.*

$G_1$ and $G_2$ in Theorem 2 could be seen as an edge perturbation of each other. Then under aforementioned mild conditions, both eigenvalues and corresponding eigenvectors of the graph are continuous. If the construction kernel $h(\cdot; \cdot)$ of LFS is smooth enough, the continuity of LFS under edge noise could, therefore, be seen as a corollary of Theorem 2.

**Corollary 1** (Continuity on Edge Noise)**.** *Let $s_i(t)$ be the LFS of node $i \in V$ as a function of $t$. Let $\mathcal{L}$ be the Laplacian of graph $G = (V, E)$, and $G' = (V, E')$ be a perturbed graph of $G$ with $\mathcal{L}' = \mathcal{L} + \Delta L$. If $\mathcal{L}$ and $\mathcal{L}'$ satisfies the conditions as in* Theorem 2*, and $h(\cdot; \cdot) \in C^2(\mathbb{R}_+^2)$, then the LFS $s_i'(t)$ of the node $i \in V$ in $G'$ satisfies*

$$|s_i'(t) - s_i(t)| \leq C_1(\delta, t)\epsilon$$

*where $C_1(\delta, t)$ is a constant independent on $\epsilon$.*

Similarly, we also prove the continuity of LFS under addition/removal of weak nodes. Weak nodes here means a set of nodes that are loosely connected to the rest of the graph.

**Corollary 2** (Continuity on Addition of Nodes)**.** *Let $s_t(i)$ be the LFS of node $i \in V$ as a function of $t$. Let $\mathcal{L}$ and $\mathcal{L}'$ be the Laplacian of $G = (V, E, w)$ and $G' = (V + \Delta V, E', w + \Delta w)$. If $\mathcal{L}$ and $\mathcal{L}'$ satisfies the conditions as in* Theorem 2*, and for some constant C,*

$$C(\sqrt{|V| + |\Delta V|})\sqrt{\sum_{i,j \in V} \Delta w_{ij}^2 + \sum_{i \in V, j \in \Delta V} w_{ij}^2} \leq \epsilon,$$

*and $h(\cdot;\cdot) \in C^2(\mathbb{R}_+^2)$, then the LFS satisfies*

$$|s_t'(i) - s_t(i)| \leq C_2(\delta, t)\epsilon$$

*where $C_2(\delta, t)$ is a constant independent on $\epsilon$.*

**Corollary 3** (Continuity on Removal of Nodes)**.** *Let $s_t(i)$ be the LFS of node $i \in V$ as a function of $t$. Let $\mathcal{L}$ and $\mathcal{L}'$ be the Laplacian of $G = (V, E, w)$ and $G' = (V - \Delta V, E', w + \Delta w)$. If $\mathcal{L}$ and $\mathcal{L}'$ satisfies the conditions as in* Theorem 2*, and for some constant C',*

$$C'(\sqrt{|V|})\sqrt{\sum_{i,j \in V - \Delta V} \Delta w_{ij}^2 + \sum_{i \in V - \Delta V, j \in \Delta V} w_{ij}^2} \leq \epsilon,$$

*and $h(\cdot;\cdot) \in C^2(\mathbb{R}_+^2)$, then the LFS satisfies*

$$|k_i'(t) - k_i(t)| \leq C_3(\delta, t)\epsilon$$

*where $C_3(\delta, t)$ is a constant independent on $\epsilon$.*

## 6.3 Matching of Noisy Graphs

We consider two graphs $G_1 = (V_1, E_1)$, and $G_2 = (V_2, E_2)$, with $|V_1| = n_1, |V_2| = n_2$. Without loss of generality, we assume $n_1 \leq n_2$. The goal is to find a one-to-one correspondence $P : V_1 \to V_2$ between $G_1$, and $G_2$, such that $v_1 \sim P(v_1)$, $\forall v_1 \in V_1$.

If $n_1 = n_2$, the problem becomes similar to graph isomorphism problem. However, with the effect of edge noise, such an isomorphism may not exist. So our goal is also not to find such an isomorphism.

If $v_1$ and $P(v_1)$ are perturbed pair, their LFS distance would be near to each other (due to the continuity property). Therefore, the similarity between $v_1$ and $P(v_1)$ in our problem is defined as the similarity in LFS distances. Then our problem could be formulated as to find $P$, such that the sum of LFS distance between the matched pairs is minimized. Rigorously,

$$P = \arg \min_{Q \in F_{n_1}} \sum d_{\text{LFS}}(v_1, Q(v_1)), \tag{13}$$

where $F_{n_1}$ is the set of all injective mappings from $V_1$ and $V_2$.

To solve this problem, we construct a bipartite graph $G = (V_1, V_2, E)$ from $G_1$ and $G_2$ with $E = V_1 \times V_2$. The optimization problem in (13) could be reformulated as the Maximum Weighted Bipartite Matching (MWBM) problem [144]. Recall that a matching of a graph $G = (V, E)$ is a subset $M$ of $E$, such that no two edges in $M$ share the same node. The MWBM problem is then to find a *matching* $M$ such that the total summed weights of the edges in $M$ is maximized. This is equivalent to our problem when we set the edge weights in $G$ as $w(e) = -d_{\text{LFS}}(e)$, for $e = (u, v)$.

The MWBM problem can be solved in running time complexity of $O(|V|^2 \log(|V|) + |V| \cdot |E|)$. In our setting, $|V| = n_1 + n_2$, and as $G$ is a complete bipartite graph, $|E| = n_1 n_2$. Therefore, the overall running time complexity is $O(n_1 n_2^2)$.

From Section 6.2.5, the optimal $P$ would have small mapped LFS distance Hence, instead of a complete bipartite graph $G$ connecting all vertices in $G_1$ with all vertices in $G_2$, we could build a sparse graph $G' = (V_1, V_2, E')$, such that we only add edges from $k$ nearest neighbors ($k$-NN) of a node in terms of the LFS distance, i.e.

$$E' = \{(i, j) \in V_1 \times V_2 \mid i \in kNN(j) \text{ or } j \in kNN(i)\}.$$

Now, $|E'| \leq k(n_1 + n_2)$, and the running time complexity is reduced to $O(n_2^2(\log n_2 + k))$. In all our experiments, we call complete bipartite graph matching *complete matching* and $k$-nearest neighbor bipartite graph matching *$k$-NN matching*.

For very large networks, we could tradeoff the solution quality for speed, and use a greedy matching algorithm. We start by randomly selecting a node in $G_1$, and search for its nearest neighbor in $G_2$ as the matched pair, and remove both from the graph and continue. This algorithm for a $k$-NN construction has linear running time.

In some cases, we know partially the identities of some of the nodes in the graphs, say the identities of big names in a coauthorship network, or groundbreaking papers in a co-citation network. We could use these disclosed identities (anchor nodes) to further narrow down the potential candidates by incorporating an additional constraint. Rigorously, for a node $u \in V_1$, find the closest anchor $s \in V_1$ (whose corresponance in $V_2$ is $s'$) and let the distance be $d^{(1)}(u, s)$ and in addition to the NN constraint, we add an edge between node $u$ and a node $u' \in V_2$ if $d^{(1)}(u, s) \geq d^{(2)}(u', s')$. We call this method pMatching, where $p$ is the percentage of anchor nodes.

## 6.4 Experiments

### 6.4.1 Matching of Perturbed Model Graphs

In this section, we use four different kind of model graphs that are widely used as models of different networks, namely, Erdos-Rényi model G$(n, m)$[3], Barabási-Albert Preferential Attachment model PAM$(n, m)$[4], Watts-Strogatz small-world model CWS$(n, k, p)$[5], and Hierarchical Network Model HNM$(n, b, c)$[6] [81]. We

[3] $G(n, m)$ is a random graph with $n$ nodes and $m$ edges uniformly randomly placed on the vertices.

[4] PAM$(n, m)$ starts from $m$ singleton nodes and add additional nodes one by one. Each newcomer connects to $m$ existing nodes with probability proportional to their current degree. The node degree in the final graph obeys a power law distribution.

[5] CWS$(n, k, p)$ starts with $n$ nodes connected to $k$ nearest neighbors on a ring topology, then each edge is rewired to a uniformly randomly selected node with probability $p$.

[6] HNM$(n, b, c)$ starts from a complete $b$-ary tree and random samples $n$ leaf nodes such that each selected node is randomly connected to $c \log^2(n)$ other nodes with replication. The probability of a connection between node $u, v$ is proportional to $f(h(u, v))$, where $f(\cdot)$ is a nonincreasing function and $h(\cdot, \cdot)$ is the distance to their common ancestor in the $b$-ary tree.

assign an exponentially distributed[7] weights on edges. To see the average matching results for different models, we generate 20 graphs with 80 nodes for each model and add random noise to the edge weights to get a perturbed counterpart. Our matching algorithm is tested and the average matching rate is recorded.

From our experiments, we discovered that $(t_{\text{min}}, t_{\text{max}}) = \log(10)(\frac{1}{\lambda_{0.9}}, \frac{1}{\lambda_{0.1}})$[8] gives the best performance for HKS, and hence is used throughout the experiments.. For WKS, we use the sampling criterion suggested in the original paper [10]. For WS, we use a kernel that is used in [59], i.e.

$$g(x) = \begin{cases} x & \text{for } x < 1 \\ -5 + 11x - 6x^2 + x^3 & \text{for } 1 \leq x \leq 2 \\ 2x^{-1} & \text{for } x > 2 \end{cases} \tag{14}$$

Throughout the experiments, we use 100 samples per signature.

From Theorem 1, we set $k = 2d$ for all kernels, and experiments showed that for Gamma, Rayleigh and t kernel, $\theta = 0.1g$ and for Inverse Chi square kernel $\theta = g$ achieve the best performance, where $g$ is the log-eigen gap, i.e. $g = \log \lambda_{\max} - \log \lambda_{\min}$.

Figures 21, 22 & 23 show the average matching performance. For all the models, matching performance drops with increasing noise level. For all signatures, their performance drops when the graph has increasing level of symmetry (or higher level of structural equivalence) – for the G$(n, m)$ model, with increasing graph density, the graph is more symmetric and hence the matching performance drops, and for PAM$(n, m)$, if the graph is too sparse, the graph is more tree-like structured, hence presents more symmetry, while for the CWS$(n, k, p)$ model, with the same rewiring probability $p$, denser graphs are more symmetry-preserved, hence lower matching rate, and with the same density, higher rewiring probability breaks the structure of the graph and makes them look more like a G$(n, m)$ graph which are known to be hard to match.

As can be seen, for all models, Gaussian(WKS)/Laplacian kernel performs the

[7] We set the scale parameter $\beta = 5$.

[8] The constant factor $\log(10)$ factor is not important in the setting, and the performance is the same when a factor of the same order of magnitude is used. $\lambda_q$ for $0 \leq q \leq 1$ is the $q$-th quantile of the eigenvalues

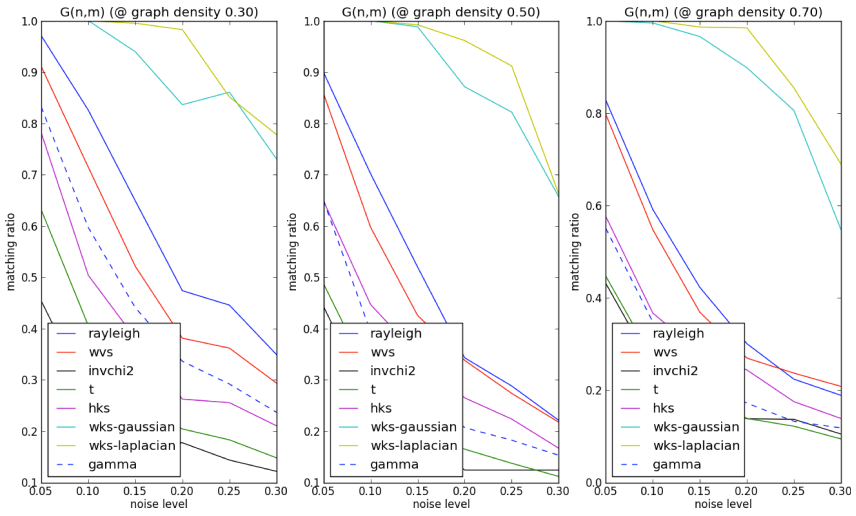



Figure 21: Average Matching Rate for Erdos-Rényi model

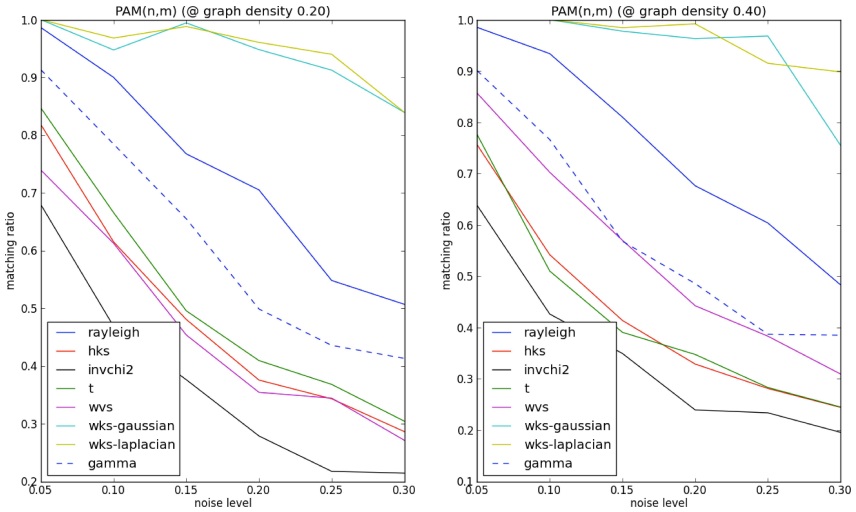



Figure 22: Average Matching Rate for Barabási-Albert Preferential Attachment model

best. One reason for this is the local property of the kernel functions. More importantly is the decoupling of the time parameter $t$ and the graph parameter $\lambda$. $\lambda$ can be seen as a natural scaling parameter of the graph, because small $\lambda$ incorporates information of a larger neighborhood of the node under the graph structure. The decoupling put more equalized weights on different natural scales, while for

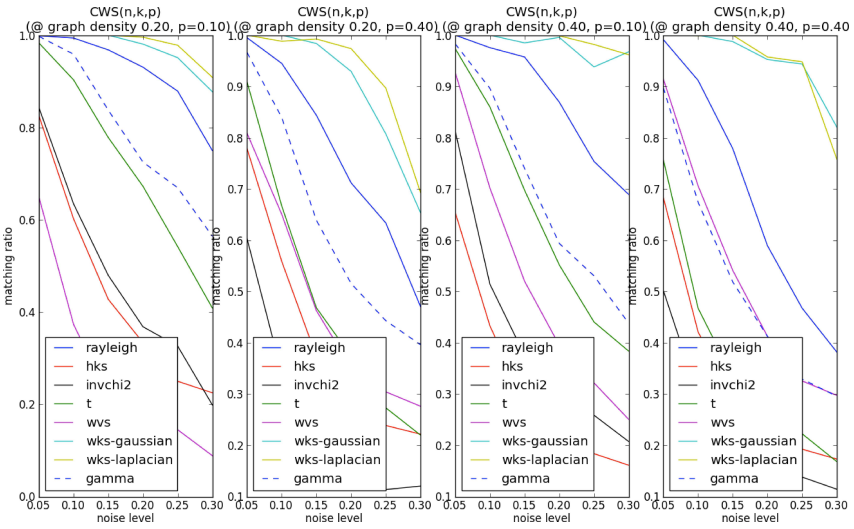



Figure 23: Average Matching Rate for Watts-Strogatz small-world model

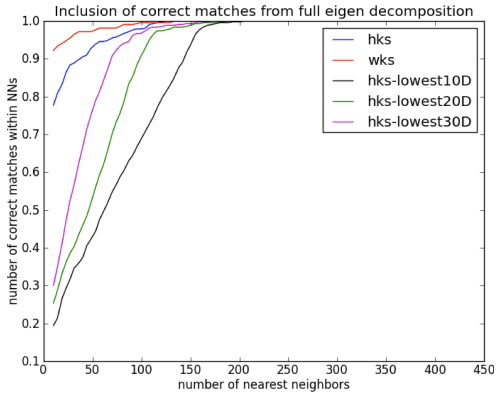
coupled kernels, larger scales are more intended to be ignored as the kernel itself can be deemed as inversely scaled by $\lambda$ which makes the sampling on most of the $t$'s under large $\lambda$ ineffective. However, one thing that is largely ignored by Gaussian/Laplacian kernels is the global property as presented by HKS. Since HKS kernel function as a special case of Gamma kernel can be seen as a low-pass filter on graph Fourier frequencies, hence putting more weights on the global information of the graph. In the following sections on real world dataset, we will focus on the comparison of WKS and HKS only and we will study their descriptive power closely.

### 6.4.2 Matching of Noisy Cocitation Networks

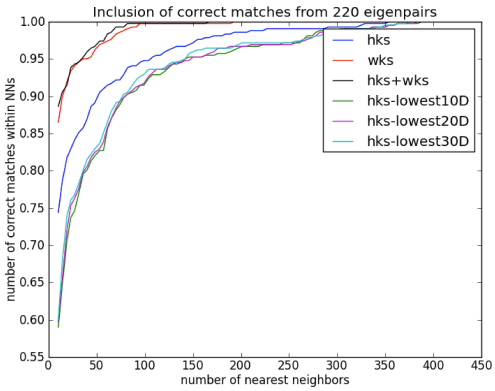
In this section, we experimented with real dataset derived from the high-energy physics theory (HEP-TH) citation network [51, 87]. In our experiments, we used the *cocitation*[9] [103] graph constructed from HEP-TH network.

We select papers of January to March 1998 from HEP-TH network as our first test nodes. To construct a small perturbation between two graphs, we randomly

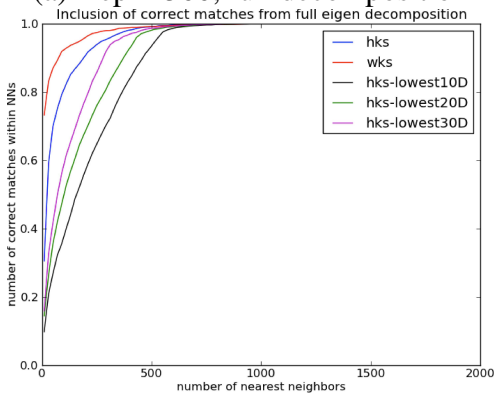
[9]The *cocitation* of two vertices $i$ and $j$ in a directed network is defined as the number of vertices that have outgoing edges pointing to both $i$ and $j$.



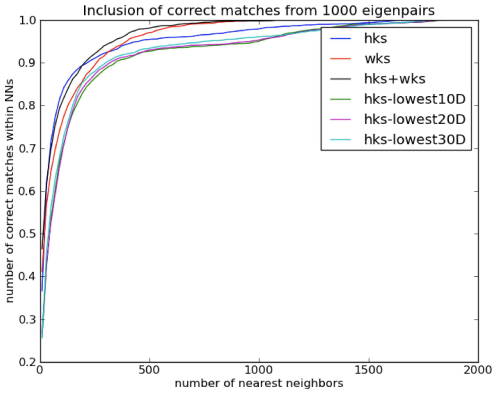

(a) HepTh500, full decomposition




(b) HepTh500, 220 eigenpairs




(c) HepTh2000, full decomposition



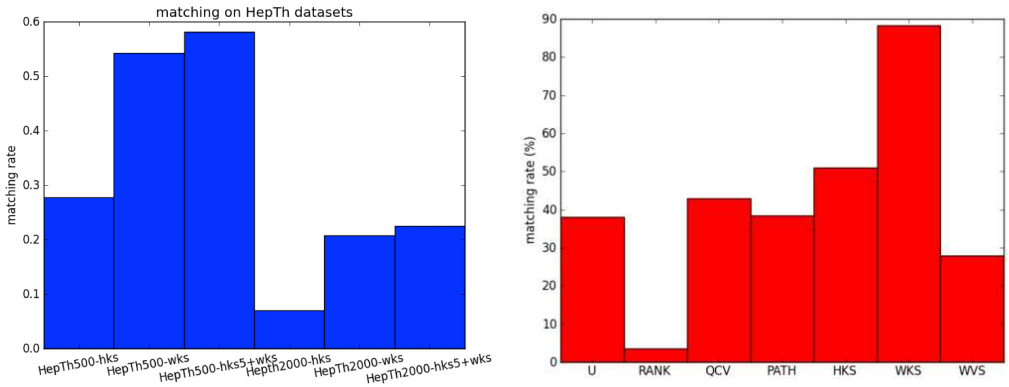

(d) HepTh2000,1000 eigenpairs

Figure 24: Inclusive rate of the HepTh500 dataset.

remove 50% of the articles from 2000 to 2002, and hence removing citation links from those papers. Edge weights are the number of cocitations. Selecting the nodes from the largest connected component, the final graphs has $|V_1| = 434, |V_2| = 436$ nodes with an overlap of 422 nodes. The weight perturbation between $G_1, G_2$ is about 5%. We call this dataset HepTh500. To increase the graph size, in the second set of test nodes, we selected nodes from the whole year of 1998 with other procedures the same, which results in graphs with $|V_1| = 2009, |V_2| = 1997$ and 1939 overlapping nodes. We call this dataset HepTh2000.

We first examine the descriptive powers of the different signatures. For that we examine for each node, whether the correct matched pair is included in its $k$-nearest neighbor, for different $k$, as shown in Figure 24. Obviously, if the correct matched

node is not even a $k$-NN, there is no way to produce the correct matching using $k$-NN matching algorithm. It can be seen for HepTh500 dataset, if we use the full eigen-decomposition to construct the signatures, WKS has the best inclusive rate. However, as the full decomposition is impractical for large scale networks, in practice only a small portion of the eigenpairs are computed to construct the signatures. In the experiment, we use both the lowest and highest portion of eigenpairs as they represent the global and local information respectively. As shown in Figure 24 (b) and (d), if we use only partially the eigenpairs, inclusive rate of WKS started to drop as well as HKS. If we combine the low-pass part of HKS and WKS as shown as *HKS+WKS* where we concatenate the first 5-D of HKS and WKS, the inclusive rate is slightly better WKS. We also plot the matching performance of each signature in Figure 25 (a). As expected, HKS+WKS gave a slightly better matching rate.




(a) Matching with partial eigen decomposition

(b) Comparison with some state-of-the-art algorithms

Figure 25: (a) Matching rate with partial eigen decomposition as described in Figure 24 (b) and (d). (b) Comparison of our algorithm with some state-of-the-art graph matching algorithm. (U) the Umeyama algorithm, (RANK) the Rank algorithm, (QCV) quadratic convex relaxation algorithm, (PATH) the PATH algorithm.

We tested the matching performance of our methods using full eigenspectrum with some of the state-of-the-art graph matching algorithms on the HepTh500 dataset as shown in Figure 25, i.e. the Umeyama algorithm [133], the Rank algorithm [122], and the QCV (quadratic convex relaxation) algorithm and the PATH algorithm [150]. The RANK algorithm is known to have a hard time to converge,

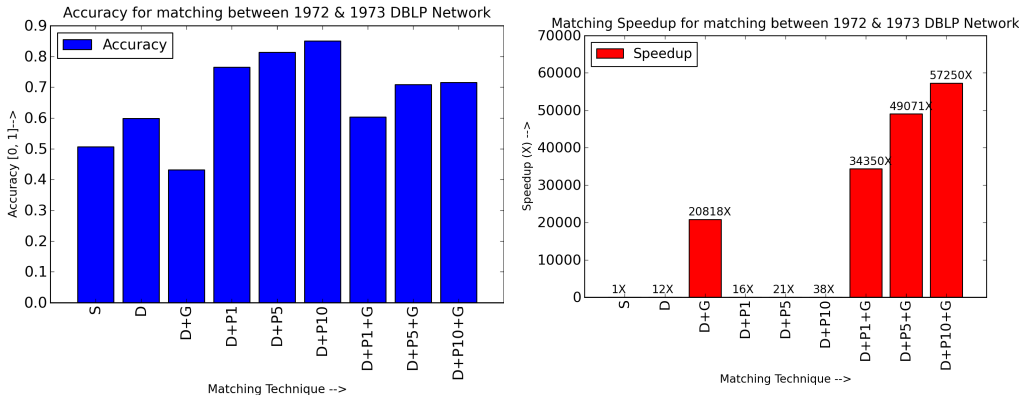
therefore, we believe at the time the results are taken, the algorithm might not have converged yet. In a nutshell, both HKS and WKS performs better, with WKS significantly outperforms the rest (achieve a matching rate of over 88%), which is consistent with previous results on random graphs.

### 6.4.3 Matching of DBLP Temporal Co-Authorship Dataset

In this section, we use the DBLP[10] data to build a co-authorship network for further evaluating our matching technique. The nodes in the network represent individual authors, while edges are papers they co-authored. As each paper is time stamped, we could utilize this temporal information to summarize the multi-edge as described in [124]. Given link weights $W_1, W_2 \ldots W_t$ at different time stamps, the exponential kernel for summarization can be computed as follows,

$$W_t^S = \begin{cases} (1-\theta)W_{t-1}^S + \theta W_t & \text{if } t > t_0 \\ \theta W_t & \text{if } t = t_0 \end{cases}$$

where $t_0$ is defined as the initial time. Thus the summarized weight gives a higher weight on recent publications and an exponentially decreasing weight on past publications. For all our experiments on the DBLP data we set $\theta = 0.25$.




(a) Matching rate

(b) Matching speed

Figure 26: Matching accuracy (left) and running time (right) for various matching schemes (*S=Standard kNN, D=Degree Heuristic, G=Greedy, P=pMatching*)

[10] `http://dblp.uni-trier.de`

In this experiment, we match summarized DBLP networks between 1972 and 1973 using WKS. The graphs have 401 and 537 nodes respectively. We set the $k$-NN to be $k = 50$ in all cases. The DBLP network is inclusive over time i.e. there are no node or edge deletions. We use this fact to further prune the potential matches by ignoring any k-NN that do not satisfy the non-decreasing degree constraint. We also consider the problem of pMatching as described earlier. In our experiments we set $p$ = 1%, 5% and 10%. To make the matching scalable for large networks we test our methods against the greedy algorithm. Figure 26 shows that we achieve a maximum accuracy of about 85%. Knowing identities to 1% of nodes helps us improve accuracy by 16%. Also Figure 26 clearly shows the speedup (>1000X) gained by using the greedy algorithm for matching by trading off with a small drop in accuracy.

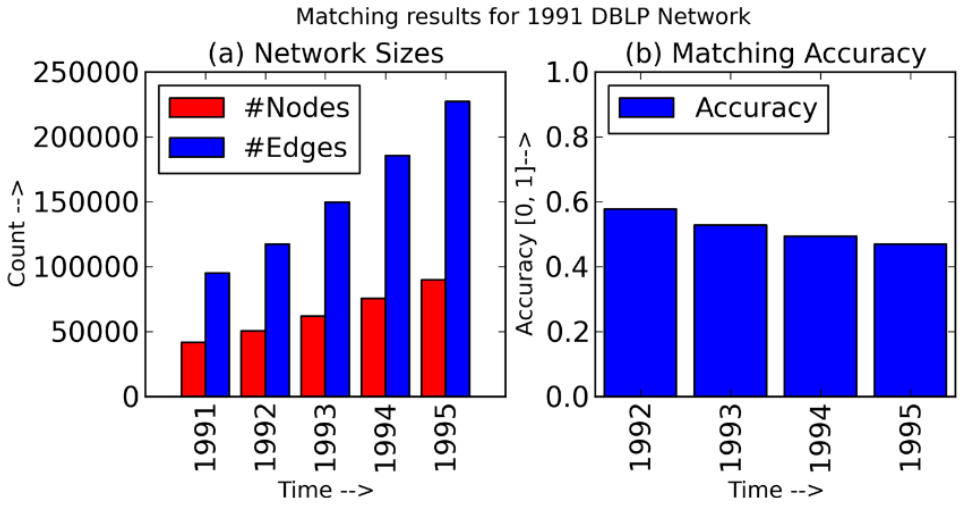



Figure 27: Matching accuracy for DBLP network between 1991 to 1995. Left: the network size; Right: the matching results.

We also consider larger DBLP networks (between the time period 1991 to 1995). We perform greedy matching with $p$=10%. Figure 27 shows the different network sizes. It also shows that we are able to achieve an accuracy of 58% for matching between the 1991 and 1992 networks. The incorrect matches are on average 2.18 hops from the actual match, which demonstrates that we are not too far away with the wrong matches. There is an expected drop in accuracy as we match the 1991 network with networks further apart in time (thus more different).

## 6.5 Conclusion and Future Work

In this chapter, we introduced the Laplacian family signatures (LFS) on graph and applied the signatures for matching noisy graphs. For a weighted graph, under some mild conditions, LFS is theoretically guaranteed to be continuous. The continuity property leads us to a matching algorithm, where we try to match nodes of a graph with those of a perturbed counterpart. By considering only the $k$-NN of nodes, our proposed algorithm achieved $O(n^2(\log n + k))$ running time complexity, where $n$ is the size of the larger graph. If we further tradeoff accuracy with speed, we also propose a simple but fast greedy algorithm to find local optima. Experiments shows WKS has the highest performance in matching within the family and across other existing state-of-the-art matching algorithms. As the whole family of signatures is far from fully explored, our future work is to investigate how to find the best kernel function. Notice that this problem is mostly data-driven and data-dependent.

# Chapter 7

# Access: News and Blog Analysis for the Social Sciences

## 7.1 Introduction

This chapter introduces the TextMap Access system which provides ready access to a wealth of interesting statistics on literally millions of people, places, and things across a number of interesting web corpora, spanning hundreds of years of documents. Powered by a flexible and scalable distributed statistics computation framework using Hadoop, currently available, continually updated corpora cover newspapers, blogs, patent records, legal documents, and scientific abstracts; well over a terabyte of raw text and growing daily. In this chapter, we briefly describe the TextMap Access system, and its impact on current research in political science, sociology, and business/marketing.

TextMap Access provides instant access to both historical and current analysis, enabling scholars, students, and the simply curious to identify cultural trends and interpret a wide range of social forces. Questions like:

- Do university reputations better track sports or academics?
- How does news sentiment and reference frequency track the winners of gubernatorial and congressional elections?
- How do changes in relational sentiment between nations correlate with trade flows, military actions, and other geopolitical events?

- Do published research results lead or lag patent trends?
- Do newly emerging news entities immediately reflect powerlaw distributions, or does it take time for such advantages to accumulate?
- Which people and places have greater (or lesser) global significance?
- How does the editorial bias of a newspaper (conservative vs. liberal) influence its coverage of events?
- What is the typical evolution of national sentiment over the course of a presidential administration?

can now be easily and meaningfully studied. Access to TextMap Access is provided at `http://www.textmap.com/access`.

The TextMap Access user interface is organized around a few principles:

**Entity orientation:** All data in Textmap is organized around named entities, as it is infeasible to provide our level of analysis for arbitrary text queries in an on-demand way.

**Graphs and data:** The most obvious products of our UI are graphic plots – time series plots, heatmaps, and network diagrams. The underlying data which generates these UI plots are accessible as comma separated value text files linked from each data page.

**We have APIs:** We have internal application programming interfaces (APIs) to provide access to our data from programs without use of the UI.

**We respect copyright:** Our system deals with text on an aggregate level, summarizing entity references on a per day, per source basis. In terms of spidering and the use of text snippets we function the same as Google News. Thus, we cannot make the text of individual articles available for redistribution.

Figure 28 is an example of one of the types of analysis Lydia makes possible. It shows sentiment subjectivity and polarity score graphs for the entity *World Trade Center* generated from a corpus of 12 U.S. newspapers, with data beginning in 1977. Subjectivity reflects the per-sentence volume of strongly positive or negative words associated with the entity, while polarity reflects the difference between positive

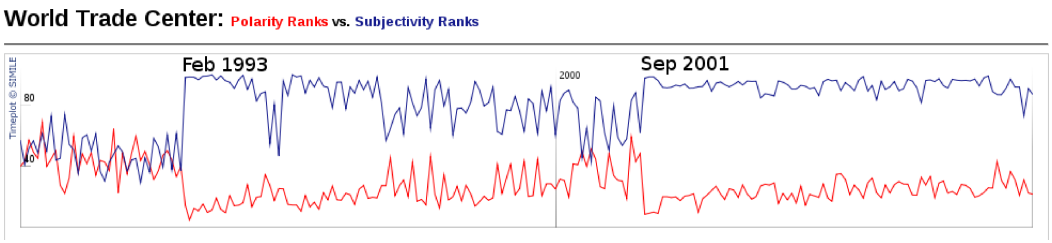



Figure 28: Sentiment subjectivity (top/blue) and polarity (bottom/red) score for the World Trade Center, reflecting both attacks on the WTC.

and negative shares of those words. It is clear from figure 28 that after both the 1993 World Trade Center bombings and the September 11th, 2001 attacks the news coverage of the *World Trade Center* became much more subjective with prevalently negative sentiment.

As another example, Table 31 shows the top entities juxtaposed with Barack Obama over four two-month periods from May to December 2008. The entities used for this experiment were taken from the list of phrases of interest to the National Annenberg Election Survey. The counts given in Table 31 are the number of sentences in which Barack Obama appeared with each respective entity. Entities having the highest juxtaposition frequencies with a given entity can thus be thought of as those "most talked about with" that entity. From Table 31 we notice that Hillary Clinton was the most associated entity with Barack Obama until she left the presidential race in June 2008, after which John McCain became his top juxtaposition. We also notice that vice-presidential candidates Joe Biden and Sarah Palin appear in Barack Obama's top juxtaposition list around the time they were announced. The word "debate" ranks fifth in the period immediately preceding the election. Finally, the White House ranks sixth on Barack Obama's top juxtaposition list in the post-election November-December 2008 period, and his top juxtaposition becomes "Democrat" instead of John McCain once the presidential race is over.

These types of studies are examples of what was only become possible with our new scalable Lydia architecture, compared to the previous version of the Lydia system described in [92]. The performance improvement with respect to computing speed is approximately 20-fold. The new Lydia system can fully process our five-year archive of over 120 million U.S. news articles in less than two weeks on our

| | May, Jun 2008 | | Jul, Aug 2008 | |
|---|---|---|---|---|
| **Rank** | **Entity Name** | **Count** | **Entity Name** | **Count** |
| 1 | Hillary Clinton | 313316 | John McCain | 358676 |
| 2 | Democrat | 346028 | Democrat | 330643 |
| 3 | John McCain | 238767 | presidential | 169928 |
| 4 | presidential | 150699 | candidate | 127159 |
| 5 | Republican | 105005 | Republican | 123940 |
| 6 | candidate | 98933 | Hillary Clinton | 103319 |
| 7 | senator | 81783 | Iraq | 74484 |
| 8 | primary | 75139 | voters | 65255 |
| 9 | voters | 75124 | Joe Biden | 64957 |
| 10 | superdelegate | 50045 | senator | 64285 |
| | **Sep, Oct 2008** | | **Nov, Dec 2008** | |
| 1 | John McCain | 563000 | Democrat | 201034 |
| 2 | Democrat | 342109 | John McCain | 178190 |
| 3 | Republican | 176279 | election | 121648 |
| 4 | presidential | 173176 | Republican | 92358 |
| 5 | candidate | 120136 | presidential | 82209 |
| 6 | Sarah Palin | 112729 | White House | 67026 |
| 7 | voters | 86988 | senate | 57876 |
| 8 | debate | 80073 | Hillary Clinton | 55913 |
| 9 | Joe Biden | 70325 | voters | 55145 |
| 10 | senator | 51129 | senator | 32575 |

Table 31: Top juxtapositions for Barack Obama for four two-month periods and the corresponding co-occurrence counts.

18-node Hadoop [8] cluster. The old Lydia ran on a single machine, but even if it could scale linearly, it would take 2.5 years to process the same dataset, according to the 250 articles per hour per machine performance estimate reported in [90].

The remainder of this chapter is organized as follows: Section 7.2 examines related text analysis systems developed for social scientists, Section 7.3 discusses the motivating social science applications of our system. Section 7.4 outlines the

features of our system's web frontend from a user's perspective, and discusses the different document streams available for analysis. Section 7.5 describes the processing phases our system takes to produce an archive of entity statistics from raw text, as well as some more technical details of the system. Finally, section 7.6 concludes the paper.

## 7.2 Prior Work

The Lydia/Textmap project has been ongoing since the Summer of 2003 [92], involving the the efforts of over thirty people over the years (mostly graduate students). The full cast of characters can be seen by clicking the `TextMap Team` link at the bottom of each Access page. The nomenclature for the Lydia/TextMap system has proven somewhat confusing:

**Lydia** refers to the analysis part of our system, the algorithms and infrastructure reducing a stream of documents to entity-oriented time series data.

**TextMap** refers to the website presenting the results of Lydia analysis on a daily news feed. In particular `www.textmap.com` collects all of our analysis on each news entity into a convenient, single-page format. Similarly, `www.textmed.com` and `www.textblg.com` provide entity analysis of Medline/Pubmed abstracts and blogs, respectively. All of these and their `.org` siblings are non-commercial websites.

**TextMap Access** , powered by a new *Lydia* analytic engine, provides dynamic access to all statistics provided by the system, queryable with a high degree of control over granularity and text sources. We will discuss these parameters in greater detail in section 7.4.

The current version of the TextMap Access interface stabilized in June 2009. Ongoing work focuses now on the implications of our analysis in areas such as political science, sociology, international relations, and business/finance; largely through our collaborations with social scientists. We are also working to develop improved analytical methods for sentiment analysis, natural language processing, and visual analytics.

The remainder of this section examines the relatively small number of other existing systems for text analysis in social sciences. These systems are largely different than Lydia in their structure and capabilities. These tools range from small-scale domain-specific semantic analysis tools (TABARI [123]), to article search tools (LexisNexis [88]), to tools for exploring structured text (Web Lab Collaboration Server [143]). Contrarily, the Lydia system provides access to statistics computed over large-scale unstructured text corpora with no human intervention.

Weigel et al. [143] propose a platform called "Web Lab Collaboration Server" to simplify large-scale web data analysis tasks for non-technical users. They try to automate tasks such as extraction of structured datasets, cleaning and formatting them. They expose a high-level interface to the user, which allows to construct extraction and analysis workflow through an intuitive GUI. The user is given primitives to construct an analysis workflow from: set operations; relational algebra; shallow text analysis, such as word count or term-frequency/inverse-document-frequency (TF-IDF); and simple graph algorithms. A user-created analysis task is converted to a logical algebra language expression, which is then compiled into map-reduce Java code and executed on a cluster. The authors use Internet Archive data in their experiments.

One primary difference between Web Lab and Lydia is that Web Lab is built around the analysis of semi-structure data, such as social network graphs or online bookstore user pages, while Lydia is designed to be capable of performing useful analysis over unstructured text, such as newspaper articles, blogs, patent records, or court decisions. Secondly, Web Lab is designed to provide users tools to create their own analysis workflows, while Lydia focuses on providing a set of pre-computed useful analyzes which are automatically updated on a daily basis, requiring no technical expertise on the part of the social-scientist using the system.

A substantial amount of work has been done in the area of automated "coding" of news sources, a task traditionally performed by undergraduate and graduate students in political science departments. News text "coding" involves finding and marking up events in news text involving pre-defined entities, such as countries or politicians. The best known systems of this kind, KEDS (Kansas Events

Data System) and its successor TABARI (Textual Analysis by Augmented Replacement Instructions) [123], developed at the University of Kansas, use a sparse parsing approach to extract event data from news text. To parse a sentence, the system marks up nouns and verb phrases in it and attempts to match them with the dictionary. TABARI is driven by a list of manually created dictionaries containing proper nouns (actors), common nouns (agents)—such as "French", verbs/verb phrases, and pronouns. Their political event coding scheme follows that of the World Event/Interaction Survey (WEIS) [96], which was later extended and superseded by the IDEA framework [19]. The TABARI and KEDS systems have been mostly used for international relations research primarily focused on the Middle East.

On the technical side, the TABARI system has been released as open source but does not contain any framework for data aggregation or parallel processing. The scale of data in studies based on TABARI such as [52] is on the order of hundreds of thousands of news reports, while our new Lydia architecture is capable of handling hundreds of millions of articles.

LexisNexis Academic [88] searches news, business, and legal content and is available to researches and students at academic institutions. Its text sources include newspapers, broadcast transcripts, blogs, SEC filings, company profiles, law reviews, case law, and statutes. The flexibility of LexisNexis search options makes it popular within the social science community. However, it offers no dedicated features for exploring a text corpus's coverage of a named entity and provides no time series or maps visualization to quantify this coverage. Here the Lydia system complements LexisNexis in a social scientist's arsenal of tools for looking at the media.

## 7.3 Applications of Lydia

The analyses provided by Lydia have already shown to be very useful for a variety of areas. In this section, we will describe a number of these applications.

### 7.3.1 Lydia in Political Science

Political science is the field that stands to most obviously benefit from using our news analysis system, as it is primarily concerned with current events involving entities widely covered in the media. To this end, we have collaborated with political scientists from Stony Brook University and University of Pennsylvania. The general direction of this collaboration is studying the influence of media coverage on electorate opinion, and our system quantifies the media part of this connection.

The Lydia project offers the opportunity to closely examine the relationship between campaign events, public opinion, and media. Many previous studies of media content such as [21, 22] have explored an extremely narrow range of news sources. With our new Lydia infrastructure we are now able to analyze roughly 1000 online news sources with an archive spanning four years, starting from November 2004, comprising over 120 million different articles. The Lydia system is able to navigate and slice the data over

- the spatial dimension (e.g. Iowa newspapers, New Hampshire newspapers or the United States as a whole), and
- the temporal dimension (on a daily, weekly, monthly, quarterly, and other scales).

Given those capabilities, the political scientists using our system in conjunction with public opinion poll data will be able to more easily answer questions such as:

- Do Iowa voters respond to sentiment expressed locally or nationally?
- Does daily media sentiment around the Iraq war influence the public opinion on the matter?
- Did media coverage of the Iraq war change after the 2004 election when Republican support for the war began to wane?

#### 7.3.1.1 Elite Influence and the Iraq War

One study conducted by political science collaborators using the Lydia system is concerned with determining the effect of elite influence on public opinion of foreign policy [69]. That is, there are essentially two models of how public opinion

of foreign policy is shaped. The first model is that public opinion is largely based upon relatively tangible, quantifiable measures (events), such as military and civilian casualties, threats to U.S. national security or strategic interest, and the overall prospect of success. The second is that public opinion is largely driven by the influence of a small number of opinion leaders or elites. The influence of elites is thought to be especially pronounced on foreign policy matters because citizens have few reliable facts or prior beliefs on which to support or oppose specific military interventions. The power of elite influence is particularly clear during times of war, when citizens typically rally around the president in support of military action.

As both events and elites are conveyed to the public through the media, it can somewhat difficult to disentangle the two effects. Huddie, Johnson, and Lebo [69] examined the effect of media tone, U.S. military casualties, Iraqi civilian casualties, specific important news events, and other effects on the partisan support for the Iraq war over a three year period. Interestingly, results indicate that the strength of these two effects differ across the political spectrum, with republicans being more responsive to specific events indicating the success of the ongoing war, while democrats were comparatively insensitive to these same events.

#### 7.3.1.2 The National Annenberg Election Survey Dataset

We have performed analysis of 16 months of news, political blog, and TV show transcript sources for the National Annenberg Election Survey (NAES), spanning the period from October 2007 to January 2009. This depository was constructed in a somewhat different manner than all our other Lydia system depositories, reflecting the specific needs of the NAES. We used a custom entity list that the NAES team provided us with, because certain words and expressions they cared about (e.g. "inexperienced" or "maverick") were not considered entities by the legacy Lydia Perl NLP pipeline.

We used documents from three classes of text: the 1000-2000 daily U.S. online newspapers we crawl on a daily basis, 45 political blogs, and transcripts of 13 political TV shows. Each transcript of a single TV show is represented as one "article" in the system.

Because the National Annenberg Election Survey team requested statistics on certain phrases that would not be normally considered entities by our NLP pipeline,

we had to implement a custom entity markup phase that would recognize entities from a pre-defined list. We were given a list of 626 phrases, which we manually grouped into synonymous phrase groups, one per line, and assigned categories to them where appropriate.

| | Daily News | | Political Blogs | | TV Transcripts | |
|---|---|---|---|---|---|---|
| 1 | John McCain | 939482 | John McCain | 9792 | Senator | 5708 |
| 2 | Democratic | 795907 | McCain | 6411 | John McCain | 3961 |
| 3 | presidential | 643408 | Hillary Clinton | 5753 | Hillary Clinton | 3842 |
| 4 | Hillary Rodham Clinton | 641815 | Democratic | 5600 | voters | 1392 |
| 5 | Democrat | 492442 | Hillary | 4895 | Democratic | 1354 |
| 6 | Hillary Clinton | 407742 | candidate | 3874 | Hillary | 1237 |
| 7 | Republican | 382197 | election | 3833 | race | 1002 |
| 8 | candidate | 333097 | voters | 3539 | McCain | 898 |
| 9 | voters | 305741 | presidential | 3191 | Democrats | 851 |
| 10 | Democrats | 237960 | Democrat | 3095 | candidate | 837 |
| 11 | race | 219677 | Democrats | 2779 | Iraq | 761 |
| 12 | primary | 213984 | race | 2292 | presidential | 712 |
| 13 | election | 203098 | black | 2242 | debate | 636 |
| 14 | Senator | 164169 | change | 2077 | election | 630 |
| 15 | black | 155659 | Iraq | 2063 | Iowa | 597 |
| 16 | senator | 142834 | Senate | 1862 | change | 581 |
| 17 | White House | 141805 | Joe Biden | 1623 | John Edwards | 576 |
| 18 | McCain | 139309 | Senator | 1615 | Democrat | 555 |
| 19 | debate | 123747 | white | 1507 | Jeremiah Wright | 543 |
| 20 | John Edwards | 103155 | John Edwards | 1158 | Pennsylvania | 491 |

Table 32: Barack Obama's top juxtapositions in the three sub-corpora of the NAES corpus.

To mark up these phrases as entities in the text, we replaced the entity markup added by our NLP pipeline with a simple dynamic programming algorithm to select a set of non-overlapping phrases in the text matching the above-mentioned predefined phrase dictionary. As the objective function in this dynamic programming algorithm we used the sum of squares of marked-up phrase lengths. This is preferable to using the sum of lengths, because when a phrase and the separate words it comprises are all included in the the custom entity dictionary, preference will be

given to marking up the full phrase, as $(a+b)^2 > a^2 + b^2$.

Table 32 offers one example of the analysis of the NAES dataset; here we subdivide into the three types of sources in our NAES dataset. Some differences between topics of political TV shows and other parts of the NAES corpus can quickly be seen:

- The word "voters" is by far the highest on the TV shows list, indicating that there is more discussion of various groups of voters, including those participating in primaries and caucuses, happening in TV shows than in other types of media.
- TV shows use the word "race" more frequently, and as it turns out looking at the TV transcripts, mostly in the "presidential race" meaning. This is an example of different language usage in different types of media.
- TV shows talk more about controversial issues, as the presence of Jeremiah Wright on the top juxtaposition list indicates.

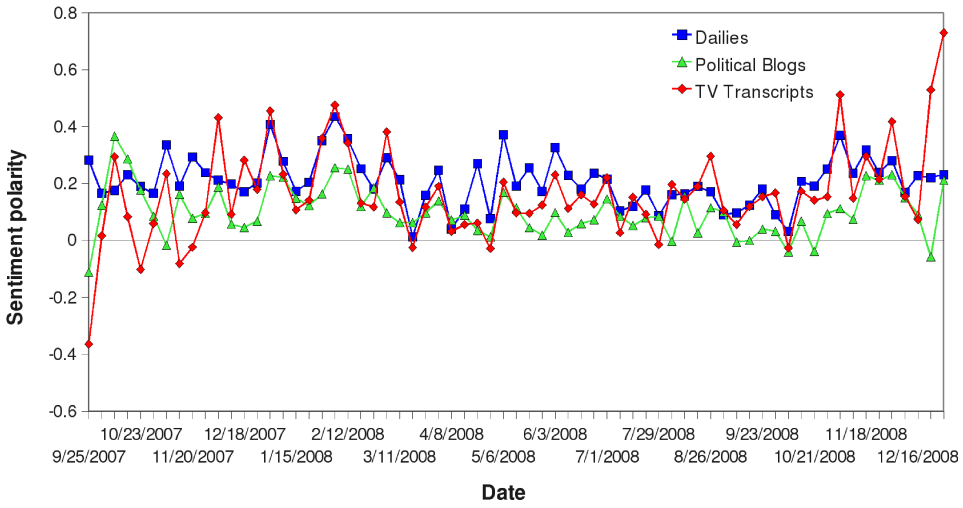



Figure 29: Weekly sentiment polarity time series for Barack Obama in the three sub-corpora of the NAES corpus.

As another example of analyzing this corpus with the Lydia system, Figure 29 shows sentiment polarity time series for Barack Obama taken from the three different parts of the corpus. The pairwise correlations between these time series

are 0.42, 0.49, and 0.37, suggesting that while they are clearly driven by a common trend, there is a difference in how these three different text sources cover events.

### 7.3.2 Lydia and Ethnic Bias in the News

Ward, Bautin, and Skiena [138] is a study of news coverage of cultural/ethnic/linguistic (CEL) groups and their interactions using the data obtained from the new Lydia system. It proposes a method for entity nationality detection using juxtaposition data, performs geographic news analysis of cultural groups, examines time series trends in CEL group frequency and sentiment, and quantifies interactions and sentiment between these groups.

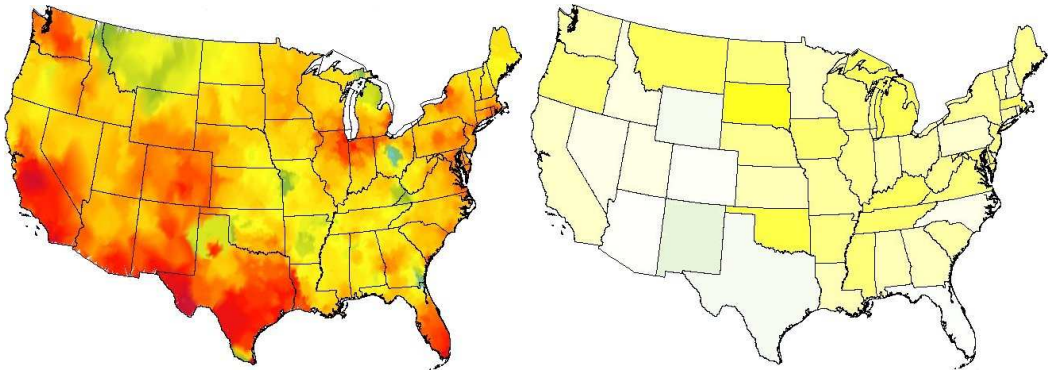


Figure 30: Frequency (left) and sentiment (right) of the Hispanic CEL group in U.S. daily newspapers. For frequency, red reflects the greatest frequency of coverage, while green reflects the least. For sentiment, yellow reflects overall positive sentiment, while white represents neutral sentiment, and green negative sentiment.

Figure 30 shows example figures from this analysis, demonstrating the geographic biases of news coverage across CEL groups. In this example, we examine the differences in volume and sentiment of coverage for individuals in the Hispanic CEL group. We note first that there is a strong regional bias to news volume, which, as we would expect, is highly correlated with the regional variation in Hispanic population density. Somewhat more interestingly, we note in the sentiment graph that overall sentiment for Hispanics is strongly inversely correlated with news volume, a trend which does not appear significantly for other CEL groups.

### 7.3.3 Lydia in Business

Zhang and Skiena [152] studies how company frequency and sentiment data obtained from the new Lydia system reflects the company's stock trading volumes and financial returns. They confirm that the news data is highly informative, as many newspaper coverage variables correlate highly with stock indicators. For example, reference volume and sentiment subjectivity correlate with trading volume, reference volume correlates with market capital, and most importantly, sentiment polarity correlates with return on investment. Using this data, Zhang and Skiena propose a news-based market-neutral trading strategy which gives consistently favorable low volatility results over a four-year period covered by our news data.

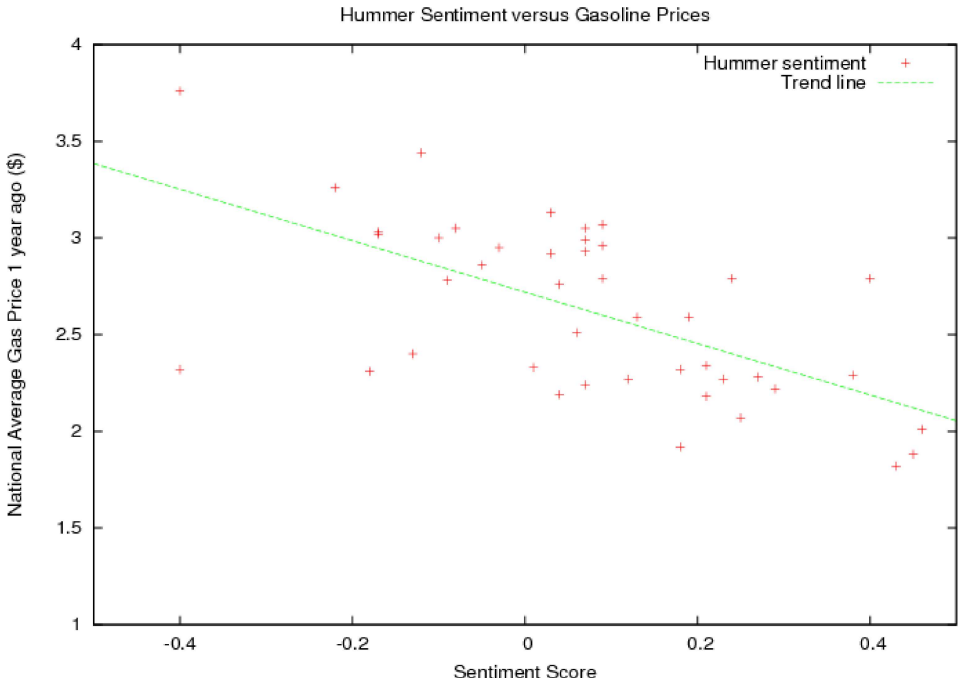



Figure 31: Comparison of Hummer Sentiment and Gas Prices.

Another area in which Lydia's analyzes can be put to good use is in the area of marketing research. One of questions that we can answer is: is the common belief that "all buzz is good buzz" actually true? Using Lydia's sentiment analysis we can decouple the volume of "buzz" surrounding a product from the sentiment polarity of the product. One example of a brand for which this belief may be in question is

the Hummer brand of vehicles. Figure 31 shows that the sentiment of newspaper coverage surrounding the Hummer brand is strongly tied (correlation 0.6) to gas prices, with a lag time of approximately one year.

### 7.3.4 Lydia in Sociology

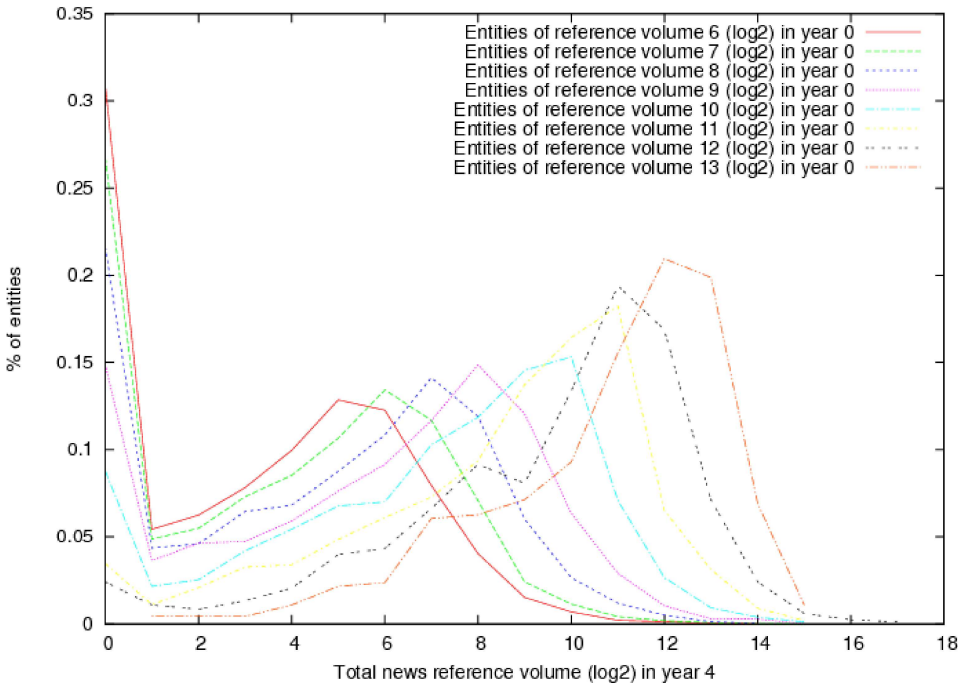



Figure 32: Reference volume distribution of news entities with broadly geographically distributed coverage after four years.

Another recent collaboration using Lydia data is in the area of sociology [134]; specifically, we are examining questions relating to the acquisition of fame. That is, questions like:

- How is fame distributed?
- Why do some people become famous, and not others? Does it pay to be a big fish in a local media market, or a lesser known national figure?
- What is the ultimate fate of news figures? Does your fate differ by your news area (sports, entertainment, etc.)?

- What role does gender play in the acquisition of fame?

Using the Lydia system, we can now begin to address questions like these, by analyzing the news records of hundreds of thousands of individuals across the country over a period of five years. As an example, Figure 32 shows the distribution of reference volume of news entities with broadly geographically distributed coverage from the beginning to the end of our U.S. dailies dataset. It is clear that the extent to which an entity is persistent in the news is strongly correlated with its initial level of coverage, and that there is a strong decay effect at work.

## 7.4 Data Analysis with Lydia

The web frontend of the Lydia system provides a way for the user to access and visualize the statistical data stored in Lydia corpora. Figure 33 shows the view of the UI. The frontend web application connects to a number of servers that are specified in its configuration, converts user input into server API requests, and displays the results of those requests to the user as graphs and tables. This provides the user with time series data of frequency, sentiment score, and relation strength to other entities for any entity or set of entities. The user can also select the source or set of sources he or she is interested in. The yielded data is available for download as a spreadsheet (.csv) as well as in graphical renderings of time series and spatial analysis ("heatmaps") of entity popularity and sentiment for any given period of time. This user interface is available at `http://www.textmap.com/access`.

Navigation within the TextMap Access webpage is organized around tabs on the grey menu bar sitting at the top of each access page, as shown at the top of Figure 33. We will now provide an overview of the general functionality of each tab, as they are organized from left to right.

**Frequency:** This tab permits you to retrieve entity statistics concerning frequency of reference on any particular entity. These counts can be aggregated over any source set, over any time scale.

**Sentiment:** This tab permits you to retrieve entity statistics concerning sentiment for any particular entity. These counts can be aggregated over any source set, over any time scale.

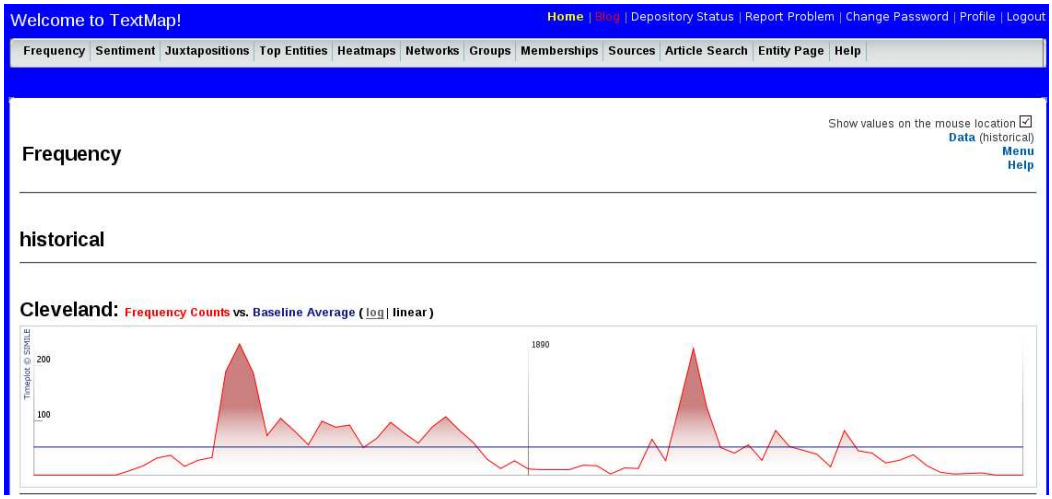



Figure 33: Lydia Web Frontend: browsing frequency time series for Grover Cleveland in the historical dataset. Cleveland was president from 1885 to 1889, and from 1893 to 1897.

**Juxtapositions:** This tab permits access to all entities collocated (or juxtaposed, or associated) with a given target entity, over any given time period. Both raw counts and a significance score are provided to measure the strength of the association. Attractive pictures of these associations are available in the networks tab. However, the juxtapositions tab is a better place to go for primary data on the strength of the underlying associations.

**Top Entities:** This tab provides access to a database defined over the most popular entities in a given depository. It is useful for identifying what is present within a given depository, to find entities worth studying and possible artifacts in our news processing.

**Heatmaps:** This tab permits access to spatial distributions of entity frequency and sentiment. These are rendered as maps showing the relative frequency interpolated over the entire United States (or the world) given the frequency observed in each geographically identified news source. Rendering interesting maps requires a source set with enough geographically distinct sources to be interesting. The dailies depository is the presently the only corpus with an interesting variety of geographically tagged sources, though in the future we hope to provide social media with

interesting geographic components.

**Networks:** This tab provides access to the network of associations centered around a given entity. The network images provide a good overview of the interactions of the given entities, and is sometimes quite revealing. The edges in this network are defined by the same criteria as in the juxtapositions tab, which is a better place to go for primary data which captures the strength of association, and which can more easily be modulated over time and source constraints.

**Groups:** This tab allows the user to create their own entity lists, over which statistics can be aggregated.

**Memberships:** This tab provides access to predefined synsets and groups associated with a given entity. A particular news entity is often referred to under several different names, such as Hillary Clinton and Hillary Rodham Clinton. Synsets are computed entity groups designed to capture all aliases associated with a particular entity. Past studies using the Lydia system have created groups of entities corresponding to ethnicity and nationality. In the future, entity classes corresponding to actors, athletes, etc. may be provided.

**Sources:** This tab permits the user to create custom subsets of media sources to be analyzed together. Suppose you have a lists of liberal and conservative newspapers. Here is where you can register these two sets – and then look for statistics extracted just from them. Custom source sets are a relatively advanced feature, so know the power exists but play with them later.

Textmap provides access to these entity statistics drawn from several different large-scale text corpora. Different insights follow from viewing entities under different corpora. Below, we summarize the analysis depositories currently available:

**Dailies:** This depository is constructed from a large corpus (over one terabyte) of U.S. and international English-language newspapers, starting November 2004. Typically, we obtain news from about 500 different sources each day, so this corpus provides a diverse spectrum of views covering a wide variety of geographic locations. It provides our most detailed and comprehensive news analysis, and is the default depository for all tabs of the Access interface.

**NAES 2008:** This is the depository of customized analysis for the 2008 *National Annenberg Election Survey*, covering the period from October 1, 2007 to January 30, 2009. It provides detailed data on a custom set of roughly 700 entities of particular relevance to the presidential election campaign, in daily newspapers, political blogs, and selected television shows. Predefined source sets can be registered to distinguish between news, blogs, and TV sources.

**Archival:** This depository is constructed from a longer-term corpus of articles over ten major U.S. daily newspapers, where each source goes back to at least 1995. The coverage is considerably thinner than the Dailies depository, but the historical sweep is considerably more interesting as it covers roughly six times as long a period.

**Historical:** This depository provides analysis from a select set of very long-range news sources (currently *The New York Times* and *Time Magazine*) providing from 1851 until present times. Because much of this text is extracted from text snippets instead of full articles, our coverage is even thinner than the previous Archival repository but the history very interesting. For example, it is instructive to view how sentiment evolves regarding entities like *Hitler* and *Stalin* before, during, and after World War II.

**Pubmed:** This depository of over 17 million Medline/Pubmed journal abstracts permits analysis of trends regarding scientific and medical research, with comprehensive coverage since 1975 and sparser coverage back to 1865.

**Patents:** This depository of over 3.7 million U.S. patent abstracts charts the scientific and technical landscape since 1971.

**Supreme Court Decisions:** The depository provides an analysis of all of the almost 60,000 U.S. Supreme Court decisions from colonial times through 2005.

**LiveJournal:** These blogs were the subject of our original study of blogs [91], and are particularly interesting as a study of randomly-selected blog postings as opposed to blogs with a significant audience.

The typical use case for the web frontend of our system is as follows. After logging into the system the user can choose,

- *The data type to access* – The major data types available through the navigation bar shown at the top of Figure 33 are entity frequency time series, entity sentiment time series, top juxtaposition lists, most popular entities, heatmaps, and entity relation networks.
- *An entity* – Even if the precise entity name entered is not present in the depository, a suggestion list is provided using the depository server's capability to search entity names.
- *A corpus or a source set within a corpus to use* – The user can define his or her own source sets on country, state, and individual source granularity, and save them for later use.
- *A time period* – (start and end date).
- *Data aggregation time scale* – (daily, weekly, monthly) and the aggregation period length (e.g. 7 to get a weekly time scale instead of daily).

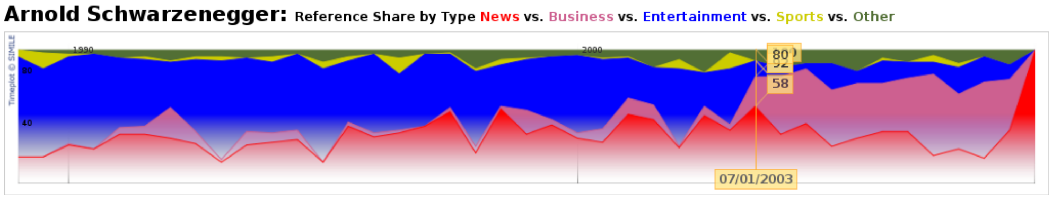



Figure 34: The distribution of occurrences of Arnold Schwarzenegger between article categories over time in an archival corpus of U.S. newspapers. The sharp increase in the fraction of "business" articles and decrease in the fraction of "entertainment" articles happen around the time he is elected the Governor of California.

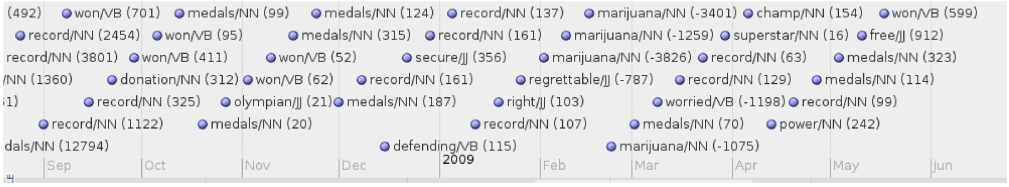



Figure 35: A sentiment word timeline for Michael Phelps.

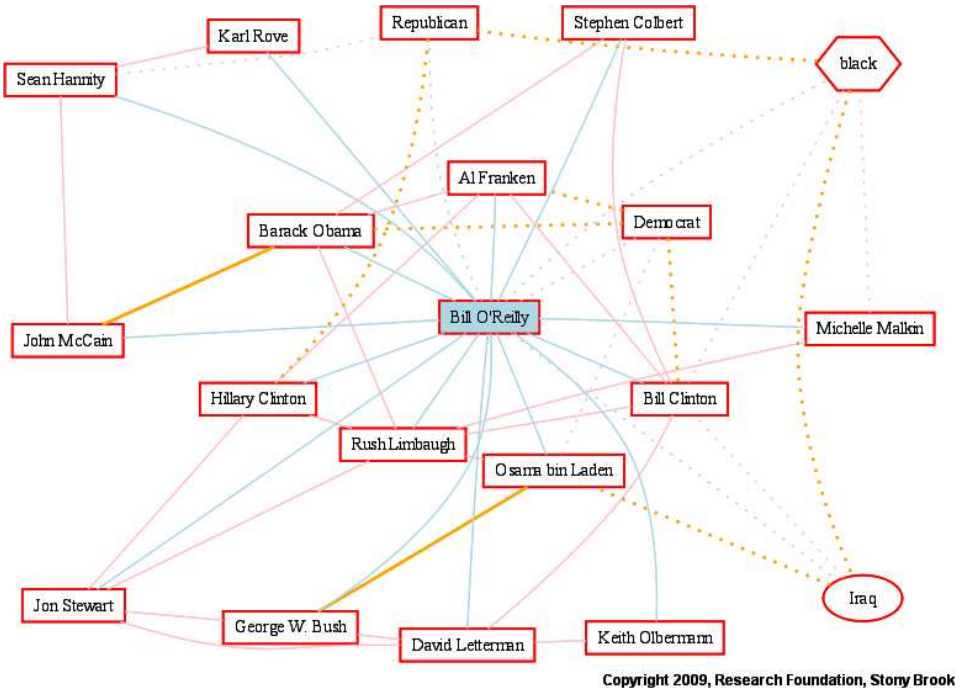



Figure 36: An entity relation network for Bill O'Reilly from the NAES 2008 corpus.

Once these parameters have been identified, the user can do the following, depending on the data access tab chosen:

- Browse time series graphs and download these time series as .csv files:
  - Entity reference frequency. For example, Figure 33 shows a time series of references for Grover Cleveland. We can also obtain the number of references across various article categories such as news, business, entertainment etc. Figure 34 shows the time series of references for Arnold Schwarzenegger across the 5 article categories.
  - Number of articles referencing the entity.
  - Number of sentences referencing the entity.
  - Sentiment counts (positive and negative), scores, and ranks.
  - A timeline of sentiment words contributing the most to an entity's sentiment score (e.g., Figure 35). This is useful for understanding the reasons of significant entity sentiment score changes.

- View the top juxtapositions for an entity, or juxtaposition time series for a given pair of entities.
- Browse the entity relation network based on entity juxtapositions, as shown in Figure 36.

## 7.5 Processing Flow

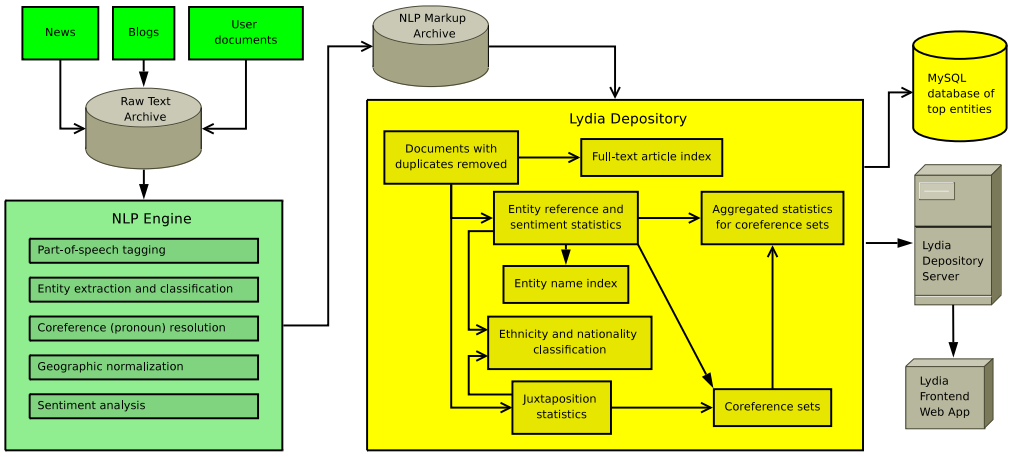



Figure 37: High-level Lydia architecture diagram

Lydia consists of five primary components: spidering, NLP markup, sentiment analysis, entity analysis and aggregation, and database export and visualization. Figure 37 shows the high-level architecture diagram of the Lydia system. We will briefly discuss each of these in turn.

**Spidering:** Lydia spiders text sources ranging from mainstream news sources to blogs on a continual, daily basis. Our largest spidered corpus consists of more than a hundred-million news articles spanning nearly five years, from over a thousand newspapers across the country and around the world. Other corpora include text derived from Lydia's own blog spiders, as well as blog text from blog-aggregation services such as Spinn3r.

**NLP Markup:** Starting from raw unstructured text, Lydia performs a series of natural language processing tasks. Lydia begins by applying a part-of-speech tagger,

the output of which is used in a number of later stages. Next, the system performs a sequence of steps designed to identify and classify named entities. Entities are classified into a range of over 100 different categories, such as *PERSON*, *COMPANY*, *ADDRESS*, or *BODY_OF_WATER*. With entities marked up, the system then attempts to perform pronoun resolution, local entity co-reference, and geographic normalization. Pronoun resolution simply means that, wherever possible, the system will attempt to resolve pronoun references to the indicated entity. Local entity co-reference is similar, but refers to resolving references to the same entity under different names. Later phases of the analysis engine, beyond the scope of discussion of this paper, attempt to unify names based on more than local context.

**Sentiment Analysis:** We refer the reader to the original papers [14, 54] for full details details of the Lydia sentiment analysis system, as well as Pang and Lee's excellent survey on techniques for sentiment analysis [110]. The *Lydia* sentiment analysis system is based on lexicons of positive and negative words, and associating entities with sentiment of co-occurring words from these lexicons. The *Lydia* sentiment lexicons were constructed by starting from small sets of seed words of incontrovertible polarity, targeted to each of six specific domains: *business*, *crime*, *health*, *politics*, *sports*, and *media*. The synonyms and antonyms of an electronic dictionary (Wordnet, [98]) enable us to expand each seed set into a full sentiment lexicon. Details of this process are reported in [54]. Although validation of sentiment analysis is a difficult problem, the accuracy and usefulness of Lydia's sentiment anaylsis has examined in several ways:

- The original *Lydia* sentiment paper [54] identifies significant correlations between our sentiment time series and political poll ratings, sports team performance, and stock indices.
- We have performed extensive studies demonstrating that *Lydia* sentiment analysis time series can be used to improve the accuracy of movie gross forecasts [151] and also as the foundation for a profitable market-neutral trading strategy for stocks [152].
- A small study comparing *Lydia* sentiment markup to human coders was performed as in the course of our involvement in the 2008 National Annenberg Election Survey. Although the human coding does not provide sentiment

markup which is ideally comparable to Lydia's, the results were still quite favorable.

**Entity Analysis and Aggregation:** Lydia takes the NLP marked-up documents and processes them in a series of jobs (virtually entirely map-reduce jobs), storing the results in a persistent data structure that we call a *depository*. A Lydia depository includes reference statistics, juxtaposition statistics, article and entity search indices, globally co-referential entity sets, a variety of derived entity classifications, and aggregated statistics for these derived groups. The new analysis architecture for Lydia is built on top of the Hadoop [8] implementation of Google's Map-Reduce [39] distributed computation model. As such, it was necessary to construct a dependency management system capable of correctly scheduling the processing of artifacts by map-reduce jobs, especially to correctly manage daily updates of newly processed text. The technical details of this are beyond the scope of this study, but can be found in [13].

**Database Export and Visualization:** Once analysis is complete, a Lydia depository can be accessed through a flexible API which exposes different slices of the data. Lydia can also export to a relational database to provide more flexibility of data exploration.

## 7.6 Conclusion

We have described a new version of the Lydia text analysis system that was designed to facilitate efficient data extraction from unstructured text, to satisfy the needs of social scientists. Our system allows a social scientist to obtain statistics about media coverage of a named entity, sentiment of this coverage, entity juxtapositions, and their changes over time. It allows to slice the news statistics by source or group of sources, time period, and time scale. These capabilities are not readily available in previous text analysis systems, which makes our system a valuable addition to a social scientist's toolbox. The new Lydia system was used to provide media coverage data of the 2008 Presidential Election to the National Annenberg Election Survey.

# Bibliography


[1] Churn rate. http://en.wikipedia.org/wiki/Churn_rate.

[2] Word-of-mouth the most powerful selling tool: Nielsen global survey. http://www.nielsen.com/us/en/insights/press-room/2007/Word-of-Mouth-the-Most-Powerful-Selling-Tool-Nielsen-Global-Survey.html.

[3] Game changer: Cone survey finds 4 out of 5 consumers reverse purchase decisions based on negative online reviews. http://www.conecomm.com/contentmgr/showdetails.php/id/4008, 2011.

[4] M. Ahmad, B. Keegan, S. Sullivan, D. Williams, J. Srivastava, and N. Contractor. Illicit bits: Detecting and analyzing contraband networks in massively multiplayer online games. *Proceedings of SocialCom-11*, 2011.

[5] M. Ahmad, B. Keegan, D. Williams, J. Srivastava, and N. Contractor. Trust amongst rogues? a hypergraph approach for comparing clandestine trust networks in mmogs. *Proceedings of Fifth International AAAI Conference on Weblogs and Social Media (ICWSM 2011)*, 2011.

[6] L. Akoglu, R. Chandy, and C. Faloutsos. Opinion fraud detection in online reviews by network effects. In *To appear in 2013 International AAAI Conference on WebBlogs and Social Media*, 2013.

[7] H. Almohamad and S. Duffuaa. A linear programming approach for the weighted graph matching problem. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 15(5):522–525, may 1993.

[8] Apache Software Foundation. The Hadoop Project. `http://lucene.apache.org/hadoop`.

[9] C. Asavathiratham, S. Roy, B. Lesieutre, and G. Verghese. The influence model. *Control Systems, IEEE*, 21(6):52 –64, December 2001.

[10] M. Aubry, U. Schlickewei, and D. Cremers. The wave kernel signature: A quantum mechanical approach to shape analysis. In *IEEE International Conference on Computer Vision (ICCV) - Workshop on Dynamic Shape Capture and Analysis (4DMOD)*, 2011.

[11] L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 181–190, New York, NY, USA, 2007. ACM.

[12] L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. Group formation in large social networks: membership, growth, and evolution. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '06, pages 44–54, New York, NY, USA, 2006. ACM.

[13] M. Bautin. *News Analysis for the Social Sciences*. PhD thesis, Stony Brook University, August 2009.

[14] M. Bautin, L. Vijayarenu, and S. Skiena. International Sentiment Analysis for News and Blogs. In *Proc. of the International Conference on Weblogs and Social Media*, Seattle, WA, April 2008.

[15] M. Bautin, C. B. Ward, A. Patil, and S. S. Skiena. Access: news and blog analysis for the social sciences. In *Proceedings of the 19th International Conference on World Wide Web*, pages 1229–1232. ACM, 2010.

[16] I. Benjamini and L. Lovász. Global information from local observation. In *Proceedings of the 43rd Symposium on Foundations of Computer Science*, FOCS '02, pages 701–710, Washington, DC, USA, 2002. IEEE Computer Society.

[17] T. Biyikoğlu, J. Leydold, and P. F. Stadler. Laplacian eigenvectors of graphs(perron-frobenius and faber-krahn type theorems). *Lecture notes in mathematics*, 2007.

[18] V. D. Blondel, A. Gajardo, M. Heymans, P. Senellart, and P. V. Dooren. A measure of similarity between graph vertices: Applications to synonym extraction and web searching. *SIAM Rev.*, 46:647–666, April 2004.

[19] D. Bond, J. Bond, C. Oh, J. C. Jenkins, and C. L. Taylor. Integrated Data for Events Analysis (IDEA): An Event Typology for Automated Events Data Development. *Journal of Peace Research*, 40:733–745, Nov 2003.

[20] Z. Borbora, K. Hsu, J. Srivastava, and D. Williams. Churn prediction in mmorpgs using player motivation theories and ensemble approach. *Proceedings of SocialCom-11*, 2011.

[21] J. Box-Steffensmeier, D. Darmofal, and C. Farrell. The endogenous relationship of campaign expenditures, expected vote, and media coverage. In *American Political Science Association annual meeting*, 2005.

[22] H. Brandenburg. Revisiting the "Liberal Media Bias": A Quantitative Study into Candidate Treatment by the Broadcast Media During the 2004 Presidential Election Campaign. In *Proc. of the Annual Meeting of the American Political Science Association*, Philadelphia, Sep 2006.

[23] O. Brdiczka, J. Liu, B. Price, J. Shen, A. Patil, R. Chow, E. Bart, and N. Ducheneaut. Proactive insider threat detection through graph learning and psychological context. In *2012 IEEE Symposium on Security and Privacy Workshops (SPW)*, pages 142–149. IEEE, 2012.

[24] L. Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[25] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[26] M. Burke and R. Kraut. Mopping up: modeling wikipedia promotion decisions. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, pages 27–36. ACM, 2008.

[27] R. Burt. *Structural holes: The social structure of competition*. Harvard University Press, 1995.

[28] D. Cartwright and F. Harary. Structural balance: a generalization of heider's theory. *Psychological review*, 63(5):277, 1956.

[29] V. R. Carvalho and W. W. Cohen. Learning to extract signature and reply lines from email. In *Conference on Email and Anti-Spam (CEAS-04)*, 2004.

[30] D. Centola and M. Macy. Complex Contagions and the Weakness of Long Ties. *American Journal of Sociology*, 113(3):702–734, Nov. 2007.

[31] W. W. Cohen, V. R. Carvalho, and T. M. Mitchell. Learning to classify email into speech acts. In *Conference on Empirical Methods in Natural Language Processing (EMNLP-04)*, 2004.

[32] J. Coleman. Social capital in the creation of human capital. *American journal of sociology*, pages 95–120, 1988.

[33] J. Coleman. *Foundations of social theory*. Belknap Press, 1994.

[34] T. R. Corporation. Factbox: A look at the $65 billion video games industry. `http://uk.reuters.com/article/2011/06/06/us-videogames-factbox-idUKTRE75552I20110606`, 2011.

[35] K. Coussement and D. V. den Poel. Churn prediction in subscription services: An application of support vector machines while comparing two parameter-selection techniques. *Expert Systems with Applications*, 34(1):313 – 327, 2008.

[36] D. R. Cox and D. Oakes. *Analysis of survival data*, volume 21. Chapman & Hall/CRC, 1984.

[37] K. Dasgupta, R. Singh, B. Viswanathan, D. Chakraborty, S. Mukherjea, A. A. Nanavati, and A. Joshi. Social ties and their relevance to churn in mobile telecom networks. In *EDBT*, pages 668–677, 2008.

[38] J. Davis. Structural balance, mechanical solidarity, and interpersonal relations. *American Journal of Sociology*, pages 444–462, 1963.

[39] J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters. In *Proc. of the OSDI'04: Sixth Symposium on Operating System Design and Implementation*, pages 137–150.

[40] W. Dong, B. Lepri, A. Cappelletti, A. S. Pentland, F. Pianesi, and M. Zancanaro. Using the influence model to recognize functional roles in meetings. *Proceedings of ICMI'07*, pages 271–278, 2007.

[41] G. Dror, D. Pelleg, O. Rokhlenko, and I. Szpektor. Churn prediction in new users of yahoo! answers. In *Proceedings of the 21st international conference companion on World Wide Web*, WWW '12 Companion, pages 829–834, New York, NY, USA, 2012. ACM.

[42] T. DuBois, J. Golbeck, and A. Srinivasan. Rigorous probabilistic trust-inference with applications to clustering. In *Web Intelligence and Intelligent Agent Technologies, 2009. WI-IAT'09. IEEE/WIC/ACM International Joint Conferences on*, volume 1, pages 655–658. IET, 2009.

[43] T. DuBois, J. Golbeck, and A. Srinivasan. Predicting trust and distrust in social networks. In *SocialCom/PASSAT*, pages 418–424, 2011.

[44] N. Ducheneaut, N. Yee, E. Nickell, and R. Moore. Alone together? exploring the social dynamics of massively multiplayer online games. *Proceedings of CHI 2006*, pages 407–416, 2006.

[45] N. Ducheneaut, N. Yee, E. Nickell, and R. J. Moore. The life and death of online gaming communities: A look at guilds in world of warcraft. *Prof. of CHI*, 2007.

[46] D. Easley and J. Kleinberg. *Networks, crowds, and markets : reasoning about a highly connected world*. Cambridge University Press, July 2010.

[47] S. Feng, L. Xing, A. Gogar, and Y. Choi. Distributional footprints of deceptive product reviews. In *Proceedings of the 2012 International AAAI Conference on WebBlogs and Social Media, June*, 2012.

[48] J. B. Ferreira, M. Vellasco, M. A. Pacheco, and C. H. Barbosa. Data mining techniques on the evaluation of wireless churn. In *In ESANN*, pages 483–488, 2004.

[49] D. Fields, M. E. Dingman, P. M. Roman, and T. C. Blum. Exploring predictors of alternative job changes. *Journal of Occupational and Organizational Psychology*, 78(1):63–82, 2005.

[50] T. Gartner, P. Flach, and S. Wrobel. On graph kernels: Hardness results and efficient alternatives. In *Proceedings of the Conference on Learning Theory (COLT)*, 2003.

[51] J. Gehrke, P. Ginsparg, and J. Kleinberg. Overview of the 2003 kdd cup. *ACM SIGKDD Explorations Newsletter*, 5(2):149–151, 2003.

[52] D. J. Gerner, R. Abu-Jabr, P. A. Schrodt, and O. Yilmaz. Conflict and Mediation Event Observations (CAMEO): A New Event Data Framework for the Analysis of Foreign Policy Interactions.

[53] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[54] N. Godbole, M. Srinivasaiah, and S. Skiena. Large-Scale Sentiment Analysis for News and Blogs. In *Proc. of the International Conference on Weblogs and Social Media*, Mar. 2007.

[55] J. Golbeck and J. Hendler. Inferring binary trust relationships in web-based social networks. *ACM Transactions on Internet Technology (TOIT)*, 6(4):497–529, 2006.

[56] M. Granovetter. The strength of weak ties. *American journal of sociology*, pages 1360–1380, 1973.

[57] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *Proceedings of the 13th international conference on World Wide Web*, pages 403–412. ACM, 2004.

[58] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

[59] D. K. Hammond, P. Vandergheynst, and R. Gribonval. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*, 30(2):129–150, Mar. 2011.

[60] C. Hart, J. Heskett, W. Sasser, et al. The profitable art of service recovery. *Harvard business review*, 68(4):148–156, 1990.

[61] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, 2001.

[62] R. Heeks. Current analysis and future research agenda on gold farming: Real world production in developing countries for the virtual economies of online games. *Institute for Development Policy and Management, University of Manchester*, 2008.

[63] F. Heider. Attitudes and cognitive organization. *The Journal of psychology*, 21(1):107–112, 1946.

[64] O. Herrera and T. Znati. Modeling churn in p2p networks. In *Simulation Symposium, 2007. ANSS '07. 40th Annual*, pages 33 –40, march 2007.

[65] J. Hirsch. An index to quantify an individual's scientific research output. *Proceedings of the National Academy of Sciences of the United states of America*, 102(46):16569, 2005.

[66] N. Hu, A. Patil, J. Gao, A. van de Rijt, and L. Guibas. A family of laplacian based descriptors for noisy graph matching. Technical report, Stanford University and Stony Brook University, 2012.

[67] B. Huang, B. Buckley, and T. M. Kechadi. Multi-objective feature selection by using nsga-ii for customer churn prediction in telecommunications. *Expert Syst. Appl.*, 37(5):3638–3646, May 2010.

[68] B. Q. Huang, M.-T. Kechadi, and B. Buckley. Customer churn prediction for broadband internet services. In *Proceedings of the 11th International Conference on Data Warehousing and Knowledge Discovery*, DaWaK '09, pages 229–243, Berlin, Heidelberg, 2009. Springer-Verlag.

[69] L. Huddie, C. Johnston, and M. Lebo. Elite influence, media coverage, and public opinion on the iraq war. In *Midwest Political Science Association 67th Annual National Conference*, 2009.

[70] S.-Y. Hung, D. C. Yen, and H.-Y. Wang. Applying data mining to telecom churn management. *Expert Systems with Applications*, 31(3):515 – 524, 2006.

[71] P. Jaccard. *Etude comparative de la distribution florale dans une portion des Alpes et du Jura*. Impr. Corbaz, 1901.

[72] G. Jeh and J. Widom. Simrank: a measure of structural-context similarity. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 538–543, New York, NY, USA, 2002. ACM.

[73] K. Jiang, D. Liu, P. F. McKay, T. W. Lee, and T. R. Mitchell. When and how is job embeddedness predictive of turnover? a meta-analytic investigation. 2012.

[74] N. Jindal and B. Liu. Opinion spam and analysis. In *Proceedings of the international conference on Web search and web data mining*, pages 219–230. ACM, 2008.

[75] S. Kairam, D. Wang, and J. Leskovec. The life and death of online groups: Predicting group growth and longevity. In *Proceedings of the fifth ACM international conference on Web search and data mining*, pages 673–682. ACM, 2012.

[76] S. Kamvar, M. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proceedings of the 12th international conference on World Wide Web*, pages 640–651. ACM, 2003.

[77] M. Karnstedt, M. Rowe, J. Chan, H. Alani, and C. Hayes. The effect of user features on churn in social networks. In *Third ACM/ICA Web Science Conference*, 2011.

[78] H. Kashima, K. Tsuda, and A. Inokuchi. Marginalized kernels between labeled graphs. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2003.

[79] J. Kawale, A. Pal, and J. Srivastava. Churn prediction in mmorpgs: A social influence based approach. In *CSE (4)*, pages 423–428, 2009.

[80] B. Keegan, M. Ahmad, D. Williams, J. Srivastava, and N. Contractor. What can gold farmers teach us about criminal networks? *ACM Crossroads*, 17(3):11–15, 2011.

[81] J. Kleinberg. Small-world phenomena and the dynamics of information. volume 14, 2001.

[82] J. Kobler, U. Schöning, and J. Toran. *The Graph Isomorphism Problem: Its Structural Complexity*. Birkhäuser Boston, 1993.

[83] C. Lampe, E. Johnston, and P. Resnick. Follow the reader: filtering comments on slashdot. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 1253–1262. ACM, 2007.

[84] E. A. Leicht, P. Holme, and M. E. J. Newman. Vertex similarity in networks. *Physical Review E*, 73(2):026120+, Feb 2006.

[85] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In *Proceedings of the 19th international conference on World wide web*, pages 641–650. ACM, 2010.

[86] J. Leskovec, D. P. Huttenlocher, and J. M. Kleinberg. Signed networks in social media. In *CHI*, pages 1361–1370, 2010.

[87] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the*

*eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 177–187. ACM, 2005.

[88] LexisNexis. LexisNexis Academic. `http://www.lexisnexis.com/us/lnacademic`.

[89] H. Liu, E. Lim, H. Lauw, M. Le, A. Sun, J. Srivastava, and Y. Kim. Predicting trusts among users of online communities: an epinions case study. In *Proceedings of the 9th ACM conference on Electronic commerce*, pages 310–319. ACM, 2008.

[90] L. Lloyd. *Lydia: A System for the Large Scale Analysis of Natural Language Text*. PhD thesis, Stony Brook University, 2006.

[91] L. Lloyd, P. Kaulgud, and S. Skiena. Newspapers vs. blogs: Who gets the scoop? In *Computational Approaches to Analyzing Weblogs (AAAI-CAAW 2006)*, volume AAAI Press, Technical Report SS-06-03, pages 117–124, 2006.

[92] L. Lloyd, D. Kechagias, and S. Skiena. Lydia: A system for large-scale news analysis. In *SPIRE*, pages 161–166, 2005.

[93] B. Masand, P. Datta, D. Mani, and B. Li. Champ: A prototype for automated cellular churn prediction. *Data Mining and Knowledge Discovery*, 3:219–225, 1999. 10.1023/A:1009873905876.

[94] P. Massa and P. Avesani. Controversial users demand local trust metrics: An experimental study on epinions.com community. In *AAAI*, pages 121–126, 2005.

[95] P. Massa and P. Avesani. Trust-aware recommender systems. In *Proceedings of the 2007 ACM conference on Recommender systems*, pages 17–24. ACM, 2007.

[96] C. McClelland. World Event/Interaction Survey (WEIS) Project, 1966-1978.

[97] G. Miller. Isomorphism testing for graphs of bounded genus. In *Proceedings of the twelfth annual ACM symposium on Theory of computing*, STOC '80, pages 225–235, New York, NY, USA, 1980. ACM.

[98] G. A. Miller. WordNet: a lexical database for English. *Commun. ACM*, 38(11):39–41, 1995.

[99] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07, pages 29–42, New York, NY, USA, 2007. ACM.

[100] K. Morik and H. Köpcke. Analysing customer churn in insurance data: a case study. In *Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases*, PKDD '04, pages 325–336, New York, NY, USA, 2004. Springer-Verlag New York, Inc.

[101] M. C. Mozer, R. Wolniewicz, D. B. Grimes, E. Johnson, and H. Kaushansky. Predicting subscriber dissatisfaction and improving retention in the wireless telecommunications industry. *Trans. Neur. Netw.*, 11(3):690–696, May 2000.

[102] A. Narayanan and V. Shmatikov. De-anonymizing social networks. In *Proceedings of the 2009 30th IEEE Symposium on Security and Privacy*, pages 173–187, Washington, DC, USA, 2009. IEEE Computer Society.

[103] M. Newman. *Networks, An Introduction*. Oxford University Press, 2010.

[104] G. Nie, G. Wang, P. Zhang, Y. Tian, and Y. Shi. Finding the hidden pattern of credit card holder's churn: A case of china. In *Proceedings of the 9th International Conference on Computational Science*, ICCS 2009, pages 561–569, Berlin, Heidelberg, 2009. Springer-Verlag.

[105] M. S. Nikulin. Chi-squared test for normality. *Proceedings of International Vilnius Conference on Probability Theory and Mathematical Statistics*, 2:119–122, 1973.

[106] J. O'Donovan and B. Smyth. Trust in recommender systems. In *Proceedings of the 10th international conference on Intelligent user interfaces*, pages 167–174. ACM, 2005.

[107] M. Ott, C. Cardie, and J. Hancock. Estimating the prevalence of deception in online review communities. In *Proceedings of the 21st international conference on World Wide Web*, pages 201–210. ACM, 2012.

[108] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock. Finding deceptive opinion spam by any stretch of the imagination. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 309–319. The Association for Computer Linguistics, 2011.

[109] W. Pan, W. Dong, M. Cebrian, T. Kim, and A. Pentland. Modeling dynamical influence in human interaction. *MIT Technical Report TR-661*, March 2011.

[110] B. Pang and L. Lee. *Opinion Mining and Sentiment Analysis*. Now Publishers, 2008.

[111] J. Parreira, D. Donato, C. Castillo, and G. Weikum. Computing trusted authority scores in peer-to-peer web search networks. In *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, pages 73–80. ACM, 2007.

[112] A. Patil, G. Ghasemiesfeh, R. Ebrahimi, and J. Gao. Social influence in epinions: A case study. Submitted to the 2013 ASE/IEEE International Conference on Social Computing (SocialCom), 2013.

[113] A. Patil, J. Liu, and J. Gao. Predicting group stability in online social networks. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 1021–1030. ACM, 2013.

[114] A. Patil, J. Liu, B. Price, H. Sharara, and O. Brdiczka. Modeling destructive group dynamics in on-line gaming communities. In *Proceedings of the 6th International AAAI Conference on WebBlogs and Social Media (ICWSM)*, pages 290–297. AAAI, 2012.

[115] A. Patil, J. Liu, J. Shen, O. Brdiczka, J. Gao, and J. Hanley. Modeling attrition in organizations from email communication. Submitted to the 2013 ASE/IEEE International Conference on Social Computing (SocialCom), 2013.

[116] M. Porter, J. Onnela, and P. Mucha. Communities in networks. *Notices of the AMS*, 56(9):1082–1097, 2009.

[117] D. Quercia, S. Hailes, and L. Capra. Lightweight distributed trust propagation. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, pages 282–291. IEEE, 2007.

[118] J. R. Quinlan. *C4. 5: programs for machine learning*, volume 1. Morgan kaufmann, 1993.

[119] J. Ramon and T. GâĹŽÃĽÂňÂğrtner. Expressivity versus efficiency of graph kernels. In *Proceedings of the International Workshop on Mining Graphs, Trees and Sequences*, 2003.

[120] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. Grouplens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186. ACM, 1994.

[121] M. Richardson, R. Agrawal, and P. Domingos. Trust management for the semantic web. *The Semantic Web-ISWC 2003*, pages 351–368, 2003.

[122] R.Singh, J.Xu, and B.Berger. Pairwise global alignment of protein interaction networks by matching neighborhood topology. *Research in Computational Molecular Biology*, 4453:16âĂŞ31, 2007.

[123] P. A. Schrodt. Automated Coding of International Event Data using Sparse Parsing Techniques. *Presented at the meeting of International Studies Association, Chicago, February*, 2001.

[124] U. Sharan and J. Neville. Temporal-relational classifiers for prediction in evolving domains. In *ICDM'08. Eighth IEEE International Conference on Data Mining, 2008*, pages 540–549. IEEE, 2008.

[125] J. Shen, O. Brdiczka, and J. Liu. Understanding email writers: Personality prediction from email messages. In *The 21st Conference on User Modeling, Adaptation and Personalization (UMAP)*, 2013.

[126] N. Shervashidze and K. Borgwardt. Fast subtree kernels on graphs. In *Advances in Neural Information Processing Systems (NIPS)*, 2009.

[127] N. Shervashidze, S. Vishwanathan, T. H. Petri, K. Mehlhorn, and K. M. Borgwardt. Efficient graphlet kernels for large graph comparison. In *Proceedings of the Artificial Intelligence and Statistics*, 2009.

[128] C. Spearman. The proof and measurement of association between two things. *The American journal of psychology*, 15(1):72–101, 1904.

[129] H. Stark and J. W. Woods. *Probability, Random Processes, and Estimation Theory for Engineers,* second edition. Upper Saddle River, NJ: Prentice Hall, 1994.

[130] J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. In *Eurographics Symposium on Geometry Processing (SGP)*, 2009.

[131] C.-F. Tsai and Y.-H. Lu. Customer churn prediction by hybrid neural networks. *Expert Syst. Appl.*, 36(10):12547–12553, Dec. 2009.

[132] J. Ugander, L. Backstrom, C. Marlow, and J. Kleinberg. Structural diversity in social contagion. *Proc. National Academy of Sciences*, 109(16):5962–5966, April 2012.

[133] S. Umeyama. An eigendecomposition approach to weighted graph matching problems. *IEEE Trans. Pattern Anal. Mach. Intell.*, 10(5):695–703, 1988.

[134] A. van de Rijt, E. Shor, C. Ward, and S. Skiena. Only 15 minutes? the social stratification of fame in printed media. *American Sociological Review*, 78(2):266–289, 2013.

[135] E. von Coelln. How big social games maintain their sticky factors. `http://www.insidesocialgames.com/2009/11/04/how-`

big-social-games-maintain-their-sticky-factors/, 2009.

[136] E. von Coelln. The sticky factor: Creating a benchmark for social gaming success. http://www.insidesocialgames.com/2009/10/27/the-sticky-factor-creating-a-benchmark-for-social-gaming-success/, 2009.

[137] J. Wang, Y. Zhang, C. Posse, and A. Bhasin. Is it time for a career switch? In *Proceedings of the 23nd International World Wide Web Conference*, pages 1377–1387. ACM, 2013.

[138] C. B. Ward, M. Bautin, and S. Skiena. Identifying differences in news coverage between cultural/ethnic groups. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 03*, WI-IAT '09, pages 511–514, Washington, DC, USA, 2009. IEEE Computer Society.

[139] S. Wasserman and K. Faust. *Social Network Analysis*. Cambridge University Press, 1994.

[140] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.

[141] D. Watts and S. Strogatz. Collective dynamics of "small-world" networks. *Nature*, 393(6684):440–442, 1998.

[142] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.

[143] F. Weigel, B. Panda, M. Riedewald, J. Gehrke, and M. Calimlim. Large-scale collaborative analysis and extraction of web data. *Proc. VLDB Endow.*, 1(2):1476–1479, 2008.

[144] D. B. West. *Introduction to Graph Theory*, volume 2. Prentice Hall Englewood Cliffs, 2001.

[145] D. R. White and K. P. Reitz. Measuring role distance: Structural, regular and relational equivalence. Technical report, University of California, Irvine, 1985.

[146] D. Williams, N. Ducheneaut, L. Xiong, Y. Zhang, N. Yee, and E. Nickell. From treehouse to barracks: The social life of guilds in world of warcraft. *Games and Culture*, 1(4):338–361, 2006.

[147] Q. Xuan, F. Du, and T.-J. Wu. Iterative node matching between complex networks. *Journal of Physics A: Mathematical and Theoretical*, 43(39):395002, 2010.

[148] Q. Xuan and T. J. Wu. Node matching between complex networks. *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, 80(2):026103+, 2009.

[149] N. Yee. The labor of fun: How video games blur the boundaries of work and play. *Games and Culture*, 1(1):68–71, 2006.

[150] M. Zaslavskiy, F. Bach, and J.-P. Vert. A path following algorithm for the graph matching problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2227–2242, 2009.

[151] W. Zhang and S. Skiena. Improving movie gross prediction through news analysis. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology - Volume 01*, WI-IAT '09, pages 301–304, Washington, DC, USA, 2009. IEEE Computer Society.

[152] W. Zhang and S. Skiena. Trading strategies to exploit blog and news sentiment. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*, pages 375–378, 2010.

[153] G. Zhao, B. Luo, J. Tang, and J. Ma. Using eigen-decomposition method for weighted graph matching. In *ICIC'07: Proceedings of the intelligent computing 3rd international conference on Advanced intelligent computing theories and applications*, pages 1283–1294, Berlin, Heidelberg, 2007. Springer-Verlag.

[154] Y. Zhao, B. Li, X. Li, W. Liu, and S. Ren. Customer churn prediction using improved one-class support vector machine. In *Proceedings of the First international conference on Advanced Data Mining and Applications*, ADMA'05, pages 300–306, Berlin, Heidelberg, 2005. Springer-Verlag.

[155] E. Zheleva, H. Sharara, and L. Getoor. Co-evolution of social and affiliation networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 1007–1016, New York, NY, USA, 2009. ACM.