Structural Change and Interaction Behavior in Multimodal Networks

Technical Report

Dr. Loo-Nin Teow, Xinghao Pan, Wen-Haw Chong, Belinda Wei-Shan Toh
DSO National Laboratories

Prof. Ee-Peng Lim, Asst. Prof. Jing Jiang, Dr. Byung-Won On
Singapore Management University

July 30, 2010

Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE **30 JUL 2010**	2. REPORT TYPE **Final**	3. DATES COVERED **17-07-2009 to 29-07-2010**

4. TITLE AND SUBTITLE **Structural Change and Interaction Behavior in Multimodal Networks**	5a. CONTRACT NUMBER **FA23860914124**
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) **Loo-Nin Teow; Xinghao Pan; Wen-Haw Chong; Belinda Wei-Shan Toh; Ee-Peng Lim**	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **DSO National Laboratories,(Singapore Management University),20 Science Park Drive,Singapore,SN,118230**	8. PERFORMING ORGANIZATION REPORT NUMBER **N/A**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) **Asian Office of Aerospace Research & Development, (AOARD), Unit 45002, APO, AP, 96338-5002**	10. SPONSOR/MONITOR'S ACRONYM(S) **AOARD**
	11. SPONSOR/MONITOR'S REPORT NUMBER(S) **AOARD-094124**

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This work presents the results of research focused on mining information from multi-network interactions for the purpose of link prediction. Multi-networks are a generalization of multimodal networks. Multi-network link prediction was evaluated on the HEP-th (theoretical high-energy physics) authorship multinetwork. Achievements include 1) a novel iterative procedure for estimating unified multinetwork node similarity based only on the network structure information; 2) label propagation algorithm to perform adjacency propagation through the similarity matrices to produce a ranking of potential new links. The work also researched modelling engagingness and responsiveness behaviors in email networks and messaging networks. Several quantitative models for measuring user engagingness and responsiveness behaviors were defined.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**64**	

Contents

Introduction

Background

This is a DSO National Laboratories project, funded by DARPA through AOARD (award number FA2386-09-1-4124), with Singapore Management University as a sub-contractor. It is a one-year project, officially starting on 14 July 2009.

Objective

The goals of this project are twofold:

1. Structural change: To perform analysis of structural change in multimodal networks across a variety of domains in a unified framework, with the eventual goal of developing a multimodal link prediction algorithm.
2. Interaction behavior: To investigate characterization and measurement of behavior multimodal networks (e.g. engagingness, responsiveness, etc.).

Achievements

Structural change:

Our research focused on mining information from multi-network interactions for the purpose of link prediction. (We view multi-networks as a generalization of multimodal networks.) We evaluated our multi-network link prediction algorithm on the HEP-th (theoretical high-energy physics) authorship multi-network. Our achievements for this part can be summarized as such:

1. We proposed a novel iterative procedure for estimating unified multi-network node similarity based only on the network structure information.
2. We extended an existing label propagation algorithm to perform *adjacency propagation* through the similarity matrices to produce a ranking of potential new links.
3. We evaluated our link prediction algorithm with the real-world HEP-th dataset, and demonstrate the ability of our algorithms in exploiting multi-network information for the purpose of improving link prediction performance.

Interaction behaviors:

For this part, we focused on modelling *engagingness* and *responsiveness* behaviors in email networks and messaging networks. The Enron email data and MyGamma Social Network Message data were used as the target datasets. Our achievements for this part can be summarized as such:

1. We defined several quantitative models for measuring user engagingness and responsiveness behaviors prevalent in email networks. We then adapted these models, and also developed new models for messaging networks.
2. We have applied the respective models to the Enron email network and MyGamma messaging network. Comparisons between engagingness and responsiveness, and comparisons between different models, were made using these real-world datasets.

3. We introduce email reply order prediction as a novel task, and show experimentally using the Enron data that the user behaviors are useful features in the prediction task.
4. We finally show that engaging and responsive users play important roles in messaging topics within the MyGamma online community, specifically, major topics in the community are driven by engaging and responsive users.

Part A: Structural Change

Multimodal Node Similarity for Link Prediction

Multi-Network Node Similarity for Link Prediction

Pan, Xinghao
DSO National Laboratories
pxinghao@dso.org.sg

Teow, Loo Nin
DSO National Laboratories
tloonin@dso.org.sg

1 Introduction

In recent years, the study of networks has been receiving a considerable amount of attention by researchers from diverse fields such as sociology, physics, biology and computer science. It is increasingly recognized that many real-world domains are highly relational in nature, as entities do not exist in isolation but constantly interact with one another.

A problem of particular importance is link prediction, specifically conjecturing the formation of new edges between entities over time. Primary applications include friend recommendation in social networks, and predicting collaboration between authors [14]. Link prediction has also been applied to market targetting [24] and movie ratings prediction [19] among other uses.

One approach to predicting links in a homophilic network [12] (e.g. friendship or co-authorship networks) would involve first computing a similarity measure between every pair of entity, and simply ranking each potential link by the pair-wise similarity value [14]. However, such methods are constrained to only single-relation homophilic networks.

On the other hand, real-world networks are highly complex, often comprising of multiple types of entities and relationships. For example, cities are linked by transportation routes in geographical networks, IP addresses are linked by LAN connections in cyber networks, and bank accounts are linked by transfers in financial networks [25]. In a complex social network, people are linked by friendships, family ties, superior-subordinate and other relationships. Furthermore, links between different types of entities (e.g. poeple living in cities and owning bank accounts) facilitate interactions between the multiple networks. Intuitively, these interactions contains additional information about the various entities, and can possibly be exploited for the purpose of link prediction.

In order to improve link prediction, we propose to analyze these interactions collectively in a *multi-network*. We then peform *adjacency propagation* to produce a ranking of potential new links. The two key intuitions behind our approach are as follows:

1. *Node similarity*: Similar nodes have common neighbors, and are linked to nodes that are themselves similar.

2. *Link preference*: A node U is more likely to form links with another node V, if V is similar to nodes to which U is linked.

These intuitions are formalized in the later sections. The first intuition is applied to estimation of multi-network node similarities; the second intuition is applied to multi-network link prediction. In using node similarities, our approach can be seen as being in the same class as the framework of Liben-Nowell & Kleinberg [14] for homophilic single-relation networks, but further improving the link prediction, and also extending to the general multi-network setting.

The following summarizes the important research contributions of our work in multi-network link prediction:

- A novel iterative procedure for estimating a *unified multi-network node similarity* based only on the network structure information.

- Extending the label propagation algorithm [30, 26] to perform *adjacency propagation* through the similarity matrices to produce a ranking of potential new links.

- Experimental results using a real-world authorship multi-network, demonstrating the ability of our algorithms in exploiting multi-network information for the purpose of improving link prediction performance.

The remainder of our report is organized as follows. We first formulate the problem of multi-network node similarity estimation for link prediction in Section 2. In Section 3, we describe our approach for both estimating node similarity and link prediction. We then show experimentally that (1) information encoded in other relations is useful for improving link prediction; and (2) our proposed method is able to exploit such information. Related work is presented before we finally conclude in Section 6.

2 Problem Formulation

2.1 Preliminaries

We begin with some definitions and notations.

A *simple network* $\mathbb{G} = \langle \mathcal{X}, \mathbf{A} \rangle$ consists of a set of nodes or entities $\mathcal{X} = \{x_1, \ldots, x_n\}$ and an adjacency function $\mathbf{A} : \mathcal{X} \times \mathcal{X} \mapsto \{0, 1\}$, such that $\mathbf{A}(x_i, x_j) = 1$ whenever there is a link from x_i to x_j. We will also treat $\mathbf{A} \in \{0, 1\}^{n \times n}$ as an adjacency matrix, such that $a_{i,j} = \mathbf{A}(x_i, x_j)$.

A *mode* [27] refers to a distinct set of entities. A *multi-modal* network consists of possibly more than one distinct set of entities (e.g. users and movies in a movie rating network).

A *relation* [27] refers to a distinct set of links. A *multi-relational* network consists of possibly more than one distinct set of links (e.g. 'is-enemy-of' and 'is-friend-of' in a social network).

More generally, a *multi-network* is a multi-modal, multi-relational network. We denote such a network with $\mathbb{G} = \langle \{\mathcal{X}_1, \ldots, \mathcal{X}_M\}, \{\mathbf{A}_{p \to q}\} \rangle$, with $\mathcal{X}_p = \{x_{p,1}, \ldots, x_{p,n_p}\}$ denoting the distinct modes, and $\mathbf{A}_{p \to q} : \mathcal{X}_p \times \mathcal{X}_q \mapsto \{0, 1\}$ denoting an adjacency function (or adjacency matrix $\mathbf{A}_{p \to q} \in \{0, 1\}^{n_p \times n_q}$) from the mode $\mathcal{X}_p$ to the mode $\mathcal{X}_q$. A simple network is thus a uni-modal, uni-relational network. A multi-network can also be seen as a composition of multiple uni-relational networks $\langle \{\mathcal{X}_p, \mathcal{X}_q\}, \mathbf{A}_{p \to q} \rangle$

A relation $\mathbf{A}_{p \to q}$ is an *undirected relation* if $p = q$ and $\mathbf{A}_{p \to q} = \mathbf{A}^T_{p \to q}$. A network $\mathbb{G} = \langle \{\mathcal{X}_1, \ldots, \mathcal{X}_M\}, \{\mathbf{A}_{p \to q}\} \rangle$ is an *undirected network* if every $\mathbf{A}_{p \to q}$ is an undirected relation.

A node $x_{q,j}$ such that $\mathbf{A}_{p \to q}(x_{p,i}, x_{q,j}) = 1$ is termed an *out-neighbor* of $x_{p,i}$. The set of nodes $\{x_{q,j} : \mathbf{A}_{p \to q}(x_{p,i}, x_{q,j}) = 1\}$ is the *out-neighborhood* of $x_{p,i}$. Conversely, $x_{p,i}$ is an *in-neighbor* of $x_{q,j}$ if $\mathbf{A}_{p \to q}(x_{p,i}, x_{q,j}) = 1$, and the set $\{x_{p,i} : \mathbf{A}_{p \to q}(x_{p,i}, x_{q,j}) = 1\}$ is the *in-neighborhood* of $x_{q,j}$. Both neighbors and neighborhoods are defined with respect to the relation $\mathbf{A}_{p \to q}$.

We further introduce a temporal aspect to the multi-network so that $\mathbb{G}_{\mathbb{t}} = \langle \{\mathcal{X}_{\mathbb{t},1}, \ldots, \mathcal{X}_{\mathbb{t},M}\}, \{\mathbf{A}_{\mathbb{t},p \to q}\} \rangle$ denotes a multi-network at time $\mathbb{t}$. The subscript $\mathbb{t}$ will be omitted when the time frame is clear from the context.

2.2 Problem definition

In the real world, multi-networks evolve structurally over time through gain and loss of both nodes and links. In this report, we are interested in the addition of new edges between existing vertices. In particular, we consider the problem of ranking potential new links for each existing node. We term this problem ***multi-network temporal link prediction***. Formally,

> Given $\mathbb{G}_{\mathbb{t}} = \langle \{\mathcal{X}_{\mathbb{t},1}, \ldots, \mathcal{X}_{\mathbb{t},M}\}, \{\mathbf{A}_{\mathbb{t},p\to q}\}\rangle$ at time $\mathbb{t}$, for each node $x_{p,i} \in \mathcal{X}_{\mathbb{t},p}$ and relation $\mathbf{A}_{\mathbb{t},p\to q}$, can we accurately rank nodes $\{x_{q,j} \in \mathcal{X}_{\mathbb{t},q} : \mathbf{A}_{\mathbb{t},p\to q}(x_{p,i}, x_{q,j}) = 0\}$ according to the likelihood that $\mathbf{A}_{\mathbb{t}+\mathbb{1},p\to q}(x_{p,i}, x_{q,j}) = 1$?

We observe that real world entities participate in various interactions with different types of entities. Intuitively, these interactions should provide us with more information about the relation for which we are performing link prediction on. For instance, consider two authors Alice and Bob who have each collaborated with the same authors. Furthermore, both Alice's and Bob's publications have often cited the same papers. With this information, we are inclined to think that Alice and Bob are "similar" in some fashion, and possibly have overlapping research interests. It does not take a great leap of imagination to then think that Alice's next publication will likely be at a conference where Bob has previously published, and vice versa.

Motivated by such intuitions, we explore the research question of whether information about different interactions can be exploited to estimate a unified similarity measure between pairs of entities of the same type, for the purpose of accurate link prediction. We term this problem ***multi-network node similarity estimation***. Formally,

> Given $\mathbb{G}_{\mathbb{t}} = \langle \{\mathcal{X}_{\mathbb{t},1}, \ldots, \mathcal{X}_{\mathbb{t},M}\}, \{\mathbf{A}_{\mathbb{t},p\to q}\}\rangle$ at time $\mathbb{t}$, can we utilize information stored in the relations $\{\mathbf{A}_{\mathbb{t},p\to q}\}$ to estimate a node similarity matrix $\tilde{\mathbf{S}}_r$ for each mode $\mathcal{X}_{\mathbb{t},r}$?

Hence, our concern is with the two-part problem of first estimating multi-network node similarities, and then applying the multi-network node similarities to the problem of temporal link prediction.

We intend for our approach to be agnostic to the semantics of entities and links. The real world identities of entities and links are withheld, and only information encoded in the multi-network structure is exploited in our approach. We also do not make any homophily assumptions. By doing so, we hope to generalize our approach to a large class of multi-networks.

We also do not utilize any non-structural features of entities in the multi-network. While we do not rule out the possibility of using non-structural features for estimating node similarities (see Section 6), we have thus far focused on the usefulness of structural features in multi-networks.

3 Approach

We break down our approach into two separate parts according to the two problems defined above. Firstly, we describe how we construct a unified similiarity matrix for each mode in a multi-network. Next, we explain how the unified similarity matrices can be used for link prediction.

3.1 Multi-Network Node Similarity

The problem of measuring node similarity in simple networks is not new. Liben-Nowell & Kleinberg [14] evaluated a number of node similarity measures for their effectiveness in link prediction. We discuss two such ideas before presenting our own balanced approach extended to the multi-networks setting.

3.1.1 Common Neighbors

A direct way of measuring similarity of two nodes in a simple network is to simply count the number of neighbors that are common to both. Formally, the common-neighbors similarity is $\mathbf{S} = \mathbf{A}\mathbf{A}^T = \mathbf{A}^T\mathbf{A}$ in an undirected network where $\mathbf{A} = \mathbf{A}^T$. (In a directed network, $\mathbf{A}\mathbf{A}^T$ would define a similarity based on common *out-neighbors*. The similarity based on common *in-neighbors* can be analogously defined as $\mathbf{A}^T\mathbf{A}$.) Although simple, the common-neighbors similiarity measure performed surprisingly well in the evaluations of [14].

It is easy to extend this model to a weighted form by introducing a weight for each node, so that the (undirected) common neighbors similarity is $\mathbf{S} = \mathbf{A}\mathbf{W}\mathbf{A}^T = \mathbf{A}^T\mathbf{W}\mathbf{A}$, where $\mathbf{W}$ is a diagonal matrix with diagonal elements $w_{i,i}$ equal to the weights of the corresponding nodes x_i. For instance, if we set $w_{i,i} = (\sum_j a_{j,i})^{-1}$, then $\mathbf{A}\mathbf{W}\mathbf{A}^T$ would define a similarity based on common out-neighbors, each inversely weighed by its number of in-neighbors. Conversely, if we set $w_{i,i} = (\sum_j a_{i,j})^{-1}$, then $\mathbf{A}^T\mathbf{W}\mathbf{A}$ would define a similarity based on common in-neighbors, each inversely weighed by its number of out-neighbors.

A shortcoming of the common neighbors method for measuring node similarity is its inability to capture relationships that may exist over multiple hops. The common neighbors method is thus unable to exploit information encoded in relations several hops away in the multi-network. (In the earlier authorship network example, Alice and Bob are similar because their publications cited the same papers; this similarity is not captured by the common neighbors method.) The next method is formulated to exactly address this problem of multi-hop similarities.

3.1.2 Recursive Neighborhood Similarity

The intuition behind recursive neighborhood similarity is that similar nodes are related to similar nodes. More precisely, nodes x_i and x_j are similar if they are linked to nodes x'_i and x'_j respectively, and x'_i and x'_j are themselves similar. The idea of recursive similarity is not new, having appeared previously in SimRank [7]. We present a matrix formulation for recursive neighborhood similarity that differs mainly from SimRank in the form of normalization used.

Let $\tilde{\mathbf{S}}$ and $\mathbf{S}$ denote the node similarity matrix and neighborhood similarity matrix respectively. In practice, for a simple network, $\mathbf{S}$ is an unnormalized or unsmoothed version of $\tilde{\mathbf{S}}$. The differentiation between the two will become clearer in the next section. Further let $\hat{\mathbf{A}}$ denote a suitably weighted adjacency matrix. We can then formalize the recursive neighborhood similarity with the equations:

$$\mathbf{S} \equiv \hat{\mathbf{A}}\tilde{\mathbf{S}}\hat{\mathbf{A}}^T \tag{1}$$

$$\tilde{\mathbf{S}} \equiv D(\mathbf{S})^{-\frac{1}{2}}\mathbf{S}D(\mathbf{S})^{-\frac{1}{2}} \tag{2}$$

where $D(\mathbf{S})$ returns a diagonal matrix with diagonal elements $d_{i,j} = \sum_i s_{i,j}$. This form of normalization has been advocated in [18, 13] for spectral clustering, and is also how the graph Laplacian is normalized in spectral graph theory [1].

We follow SimRank in proposing an iterative solution to estimating recursive neighborhood similarities:

$$\mathbf{S}^{(k)} \leftarrow \hat{\mathbf{A}}\tilde{\mathbf{S}}^{(k-1)}\hat{\mathbf{A}}^T \tag{3}$$

$$\tilde{\mathbf{S}}^{(k)} \leftarrow D(\mathbf{S}^{(k)})^{-\frac{1}{2}}\mathbf{S}^{(k)}D(\mathbf{S}^{(k)})^{-\frac{1}{2}} \tag{4}$$

where $\tilde{\mathbf{S}}^{(k)}$ and $\mathbf{S}^{(k)}$ are the node similarity matrix and neighborhood similiarity matrix computed at the kth iteration.

Note that the above formulation is based on similarity of *out-neighborhoods*. To compute the recursive in-neighborhood similarity, we would simply swap $\hat{\mathbf{A}}$ and $\hat{\mathbf{A}}^T$ in the above equations.

Although this formulation of similarity is able to capture multi-hop relationships, it was demonstrated in [14] that a link predictor based on SimRank does not perform as well as one based on common neighbors.

3.1.3 Balanced Model

In their current forms, both the common neighbors and recursive neighborhood similarity methods deal with simple networks, and neither are applicable to the multi-networks setting. We propose to combine the two in a weighted fashion and further extend to multi-networks. The key assumption of our apporach is that similar nodes have common neighbors *and* are also linked to similar nodes.

Let $\rho \in [0, 1]$ be a parameter controlling the balance between the common neighbors and recursive neighborhood models. We then define the iterative procedure for the balanced model as:

$$\begin{aligned} \mathbf{S}^{(k)} &\leftarrow \hat{\mathbf{A}}[\rho\tilde{\mathbf{S}}^{(k-1)} + (1-\rho)\mathbf{I}_n]\hat{\mathbf{A}}^T \\ &= \rho\hat{\mathbf{A}}\tilde{\mathbf{S}}^{(k-1)}\hat{\mathbf{A}}^T + (1-\rho)\hat{\mathbf{A}}\hat{\mathbf{A}}^T \qquad (5) \\ \tilde{\mathbf{S}}^{(k)} &\leftarrow D(\mathbf{S}^{(k)})^{-\frac{1}{2}}\mathbf{S}^{(k)}D(\mathbf{S}^{(k)})^{-\frac{1}{2}} \qquad (6) \end{aligned}$$

where $\hat{\mathbf{A}} = \mathbf{A}D(\mathbf{A}^T)^{-\frac{1}{2}}$ is an adjacency matrix with each node inversely weighted by the square root of its number of in-neighbors. In practice, we find that this form of weighting nodes works best.

By setting $\rho = 0$ and with an initial value of $\tilde{\mathbf{S}} = \mathbf{I}_n$, we immediately get convergence with $\mathbf{S}^{(1)} = \hat{\mathbf{A}}\hat{\mathbf{A}}^T = \mathbf{A}D(\mathbf{A}^T)^{-1}\mathbf{A}^T$. This is exactly the weighted version of the common neighbors similarity, with the diagonal weight matrix $\mathbf{W} = D(\mathbf{A}^T)^{-1}$. On the other hand, by setting $\rho = 1$, the iterative update equations reduces to those used in computing the recursive neighborhood similarity.

We now extend the balanced model to include multiple relations. First, for simplicity, for every relation $\mathbf{A}_{p\to q}$, we include the reverse relation $\mathbf{A}_{q\to p} = \mathbf{A}_{p\to q}^T$ in the multi-network. This allows us to properly account for similarity based on both in- and out-neighbours, without having to explicitly consider both directions.

Let $\mathbf{S}_{p\to q}$ be the neighborhood similarity matrix with respect to the relation $\mathbf{A}_{p\to q}$. Also, let $\tilde{\mathbf{S}}_p$ be the overall node similarity matrix for mode $\mathcal{X}_p$. We can then iteratively compute the similarity matrices

$$\mathbf{S}_{p\to q}^{(k)} \leftarrow \hat{\mathbf{A}}_{p\to q}[\rho\tilde{\mathbf{S}}^{(k-1)} + (1-\rho)\mathbf{I}_n]\hat{\mathbf{A}}_{p\to q}^T \qquad (7)$$

$$\tilde{\mathbf{S}}_p^{(k)} \leftarrow D\left(\sum_q \mathbf{S}_{p\to q}^{(k)}\right)^{-\frac{1}{2}} \left(\sum_q \mathbf{S}_{p\to q}^{(k)}\right) D\left(\sum_q \mathbf{S}_{p\to q}^{(k)}\right)^{-\frac{1}{2}} \qquad (8)$$

Essentially, the neighborhood similarity $\mathbf{S}_{p\to q}$ is computed based on the common out-neighbors and out-neighborhood similarities with respect to relation $\mathbf{A}_{p \to q}$, and the overall node similarity $\mathbf{S}_p$ is the smoothed sum of the neighborhood similarity matrices $\mathbf{S}_{p \to q}$.

The balanced model can thus be thought of as combining the common neighbors model with the recursive neighborhood similarity model, and extending the combined model to the multi-network setting.

Although matrix multiplication in general is an computationally expensive operation, we note that real-world networks tend to have sparse adjacency matrices. The sparsity of $\hat{\mathbf{A}}_{p \to q}$ can be exploited to improve the computational complexity of the iterative procedure.

We point out that it is critical to differentiate between each relation by separately computing each $\mathbf{S}^{(k)}_{p \to q}$. A naive approach of combining adjacency matrices prior to computing neighborhood similarities would possibly result in illogical similarities. For example, Alice would gain a non-zero measure of similarity with a journal she published at, through virtue of having a common neighbor in the Alice's publication at the journal.

3.2 Multi-Network Link Prediction

We now discuss our adaptation of label propagation [30, 26] for multi-network link prediction. In label propagation, class labels are propagated from labeled to unlabeled data based on similarities between data points. Given a node $x_{p,i}$, we treat its adjacent neighbors $\{x_{q,j} : \mathbf{A}_{\mathbb{t},p \to q}(x_{p,i}, x_{q,j}) = 1\}$ as labeled, and non-adjacent nodes as unlabeled, i.e. we replace labels with adjacencies. By then applying the label propagation algorithm, we essentially perform *adjacency propagation* from adjacent neighbors to non-adjacent nodes. We can then rank the non-adjacent nodes $\{x_{q,j} : \mathbf{A}_{\mathbb{t},p \to q}(x_{p,i}, x_{q,j}) = 0\}$ according to the adjacency information each $x_{q,j}$ received through the propagation.

More precisely, let us define a function $\mathbf{F}_{p \to q} : \mathcal{X}_p \mapsto (\mathcal{X}_q \mapsto \mathbb{R}^+)$ for each relation $\mathbf{A}_{p \to q}$. That is, $\mathbf{F}_{p \to q}(x_{p,i})$ is itself a function, which returns a non-negative real value for each $x_{q,j} \in \mathcal{X}_q$. We can understand each $\mathbf{F}_{p \to q}(x_{p,i})$ as a vector with each entry indicating the relative likelihood of each node $x_{q,j}$ linking to the node $x_{p,i}$. We can also represent $\mathbf{F}_{p \to q}$ as a matrix, with the (i, j)-th element equal to $\mathbf{F}_{p \to q}(x_{p,i})(x_{q,i})$.

We perform the adjacency propagation for relation $\mathbf{A}_{p \to q}$ using the iterative update equation:

$$\mathbf{F}^{(k)}_{p \to q} \leftarrow \alpha \tilde{\mathbf{S}}_q \mathbf{F}^{(k-1)}_{p \to q} + (1 - \alpha) \mathbf{A}^T_{\mathbb{t},p \to q} \qquad (9)$$

Intuitively, the neighbors of each $x_{p,i}$ are the "sources" of adjacencies (second term), which are then propagated to other similar nodes (first term).

The paramter $\alpha \in (0,1)$ controls the amount of adjacency information each node $x_{q,j}$ receives from other similar nodes. The computational complexity of the iteration is dominated by the matrix multiplication $\tilde{\mathbf{S}}_q \mathbf{F}_{p\to q}^{(k-1)}$, which in general requires $O(n_q^3)$ multiplications (or $O(n_q^{2.807})$ multiplications with the Strassen algorithm [20]). It may be possible to approximate this computation by first sparsifying $\tilde{\mathbf{S}}_q$ such that elements below a threshold are set to 0. However, this approach was not tested for this report, as we were able to complete our experiments within reasonable time.

A sufficient condition [30] for convergence of this iteration is that the eigenvalues of $\tilde{\mathbf{S}}_q$ are in $[-1,1]$ and that $0 < \alpha < 1$. Now, following the analysis of [30, 26], we define the stochastic matrix $\mathbf{P} = D(\sum_p \mathbf{S}_{q\to p})^{-1} \sum_p \mathbf{S}_{q\to p} = D(\sum_p \mathbf{S}_{q\to p})^{-\frac{1}{2}} \tilde{\mathbf{S}}_q D(\sum_p \mathbf{S}_{q\to p})^{\frac{1}{2}}$. Suppose λ and $\vec{v}$ are an eigenvalue and eigenvector pair for $\tilde{\mathbf{S}}_q$ such that $\tilde{\mathbf{S}}_q \vec{v} = \lambda \vec{v}$. Then,

$$\lambda D(\sum_p \mathbf{S}_{q\to p})^{-\frac{1}{2}} \vec{v} = D(\sum_p \mathbf{S}_{q\to p})^{-\frac{1}{2}} \tilde{\mathbf{S}}_q \vec{v} = \mathbf{P} D(\sum_p \mathbf{S}_{q\to p})^{-\frac{1}{2}} \vec{v},$$

so λ and $D(\sum_p \mathbf{S}_{q\to p})^{-\frac{1}{2}} \vec{v}$ are an eigenvalue-eigenvector pair for $\mathbf{P}$. By the Perron-Frobenius theorem, we know that $\lambda \in [-1,1]$ as an eigenvalue of stochastic matrix $\mathbf{P}$, . Since this holds true for every eigenvalue of $\mathbf{S}_{p\to q}$, all eigenvalues of $\mathbf{S}_{p\to q}$ are in $[-1,1]$.

It can be shown [26] that the iteration minimizes the cost function $\frac{1}{2}\text{trace}\left(\mathbf{F}_{p\to q}^T (\mathbf{I}_{n_q} - \tilde{\mathbf{S}}_q) \mathbf{F}_{p\to q}\right) + \frac{\gamma}{2} |\mathbf{F}_{p\to q} - \mathbf{A}_{\mathbb{t},p\to q}^T|$, where $|\cdot|$ denotes the Frobenius norm, and $\alpha = \frac{1}{1+\gamma}$. The closed form solution is $\mathbf{F}_{p\to q}^* = (1 - \alpha)(\mathbf{I}_{n_q} - \alpha \tilde{\mathbf{S}}_q)^{-1} \mathbf{A}_{p\to q}^T$ [30, 26].

4 Experiment

We evaluated our method for estimating multi-network node similarity by performing link prediction on a well-known authorship network. We used an average AUC (area under ROC curve) as our performance metric, and demonstrate that the balanced model is able to significantly outpeform a baseline model based on single-relation common neighbors.

4.1 Dataset

The base dataset that we used for evaluation is the Proximity HEP-th database [8]. The Proximity HEP-Th database is based on data from the arXiv archive and the Stanford Linear Accelerator Center SPIRES-HEP

Table 1: Extracted Relations from Proximity HEP-th dataset

Relation	Connects...	Remarks
`CoAuthorship`	`Author`↔`Author`	Undirected relation of co-authorship, equivalent to `CoAuthored`
`APublication`	`Author`→`Journal`	Derived relation of locations at which authors published
`ACitation`	`Author`→`Paper`	Derived relation of papers cited by an author
`Authorship`	`Author`→`Paper`	Equivalent to `Authored`
`Affiliation`	`Author`→`EmailDomain`	Equivalent to `EmailAffil`
`CommonTopic`	`Journal`↔`Journal`	Undirected relation, derived from topic attribute of papers
`PPublication`	`Paper`→`Journal`	Equivalent to `PublishedIn`
`PCitation`	`Paper`→`Paper`	Equivalent to `Cites`
`SubDomain`	`EmailDomain`→`EmailDomain`	Derived relation. E.g. xyz.abc.com → abc.com → com

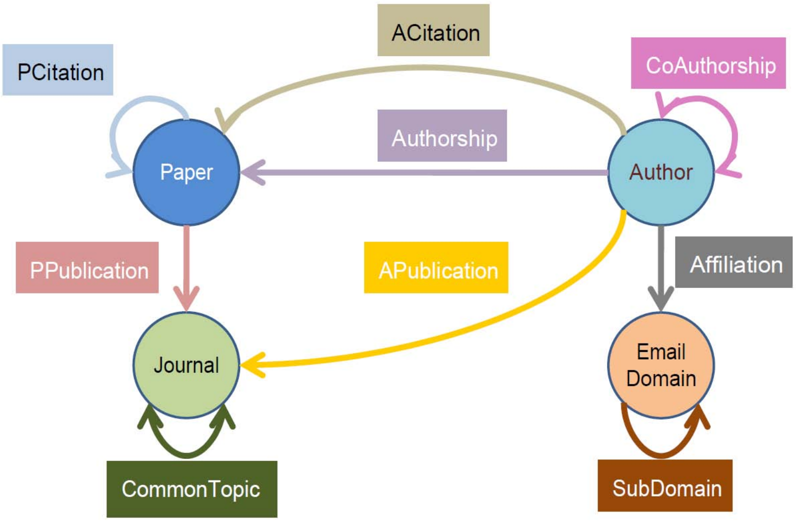


Figure 1: Graphical representation of the modified HEP-th dataset with extracted additional relations.

database provided for the 2003 KDD Cup competition with additional preparation performed by the Knowledge Discovery Laboratory, University of Massachusetts Amherst.

The dataset originally consists of four modes (`EmailDomain`, `Journal`, `Paper` and `Author`), and five relations (`PublishedIn`, `Authored`, `Cites`, `CoAuthored`, `EmailAffil`). We pre-processed the dataset to extract a total of nine relations, while maintaining the original four modes. The extracted relations are described in Table 1. Figure 1 shows a graphical representation of this extended schema.

Our intention in extracting additional relations is to create different kinds of relations. These include undirected relations (`CoAuthorship` and `CommonTopic`), static relations that do not change over time (`PCitation` and `SubDomain`), relations with tree structures (`SubDomain`), and multiple relations that connect the same pair of modes (`ACitation` and `Authorship`) but have different semantic meaning. We are therefore able to demonstrate that our solution for link prediction can generalize to different types of relations.

Although the complete dataset spanned the years 1900 till 2003, we observed that the bulk of data was concentrated in the years 1992 till 2002. Hence, we only evaluated the data within this timeframe, and segmented the data into 11 yearly time intervals.

4.2 Baseline method

We consider a baseline model which measures node similarity based only on weighted common in-neighbors. Thus, for prediction of relation $\mathbf{A}_{p\to q}$, our baseline method uses a node similarity matrix $\mathbf{S}_{q\to p} = \mathbf{A}_{p\to q}^T D(\mathbf{A}_{p\to q})^{-1}\mathbf{A}_{p\to q}$. The normalized node similarity matrix $\tilde{\mathbf{S}}_{q\to p} = D(\mathbf{S}_{q\to p})^{-\frac{1}{2}}\mathbf{S}_{q\to p}D(\mathbf{S}_{q\to p})^{-\frac{1}{2}}$ is then used for adjacency propagation, as described in Section 3.2.

Note, however, that the baseline model differs from the balanced model with $\rho = 0$ applied to the single-relation network. If the relation is directed, our balanced model considers both common in- and out-neighbors, whereas the baseline model accounts for one but not the other.

4.3 Performance metric

At each time $\mathbb{t}$, the adjacency propagation algorithm generates a matrix $\mathbf{F}_{p\to q}$ for each relation $\mathbf{A}_{\mathbb{t},p\to q}$ which provides a ranking of potential new links for each node $x_{p,i}$. However, at time $\mathbb{t}+1$, we do not observe a ranking, but a set of new-adjacencies $\mathcal{U}_{p\to q}(x_{p,i}) = \{x_{q,j} : \mathbf{A}_{\mathbb{t},p\to q}(x_{p,i}, x_{q,j}) = 0 \wedge \mathbf{A}_{\mathbb{t}+1,p\to q}(x_{p,i}, x_{q,j}) = 1\}$ and a set of non-adjacencies $\mathcal{V}_{p\to q}(x_{p,i}) = \{x_{q,j} : \mathbf{A}_{\mathbb{t},p\to q}(x_{p,i}, x_{q,j}) = 0 \wedge \mathbf{A}_{\mathbb{t}+1,p\to q}(x_{p,i}, x_{q,j}) = 0\}$.

Thus, we measure our ranking accuracy by the following performance metric:

$$acc(p \to q) = \frac{1}{|\mathcal{W}|}\sum_{x_{p,i}\in\mathcal{W}} \frac{1}{|\mathcal{U}|\cdot|\mathcal{V}|}\sum_{x_{q,j}\in\mathcal{U}}\sum_{x_{q,k}\in\mathcal{V}} \delta(\mathbf{F}(i,j), \mathbf{F}(i,k)) \qquad (10)$$

where $\mathcal{W} = \mathcal{W}_{p \to q} = \{x_{p,i} : \exists x_{q,j}, \mathbf{A}_{\mathbb{t},p\to q}(x_{p,i}, x_{q,j}) = 0 \wedge \mathbf{A}_{\mathbb{t}+1,p\to q}(x_{p,i}, x_{q,j}) = 1\}$ is the set of nodes in mode $\mathcal{X}_p$ that have acquired new adjacencies at time $\mathbb{t}+1$; $\mathcal{U} = \mathcal{U}_{p\to q}(x_{p,i})$ is the set of new-adjacencies that $x_{p,i}$ acquired at time $\mathbb{t}+1$; $\mathcal{V} = \mathcal{V}_{p\to q}(x_{p,i})$ is the set of non-adjacencies of $x_{p,i}$ at times $\mathbb{t}$ and $\mathbb{t}+1$; $\mathbf{F}(i,j) = \mathbf{F}_{p\to q}(x_{p,i}, x_{q,j})$; and

$$\delta(f_i, f_j) = \begin{cases} 1 & \text{if } f_i > f_j \\ \frac{1}{2} & \text{if } f_i = f_j \\ 0 & \text{if } f_i < f_j \end{cases}$$

When all new-adjacencies are ranked above all non-adjacencies (that is, $(\forall x_{p,i}, x_{q,j} \in \mathcal{U}, x_{q,k} \in \mathcal{V}), \mathbf{F}(i,j) > \mathbf{F}(i,k))$, the metric attains a maximum of 1, whereas when all non-adjacencies are ranked below all new-adjacencies (that is, $(\forall x_{p,i}, x_{q,j} \in \mathcal{U}, x_{q,k} \in \mathcal{V}), \mathbf{F}(i,j) < \mathbf{F}(i,k))$, the metric attains a minimum of 0. If the ranking is perfectly random and totally uncorrelated with the new-adjacencies, then the expected performance metric obtained would be 0.5.

This performance metric can also be interpreted as an AUC (area under ROC curve[1]) measure. Consider a simple threshold binary classifier $\mathbb{C}^{\tau}_{x_{p,i}}(\mathbf{F}_{p\to q}(i,j))$ which returns a predicted value of $\mathbf{A}_{\mathbb{t}+1,p\to q}(x_{p,i}, x_{q.j})$, such that

$$\mathbb{C}^{\tau}_{x_{p,i}}(\mathbf{F}_{p\to q}(i,j)) = \begin{cases} 1 & \text{if } \mathbf{F}_{p\to q}(i,j) > \tau \\ 0 & \text{otherwise} \end{cases}$$

That is, for a node $x_{p,i}$, the classifier $\mathbb{C}^{\tau}_{x_{p,i}}$ predicts a formation of a potential adjacency with node $x_{q,j}$ if and only if $\mathbf{F}_{p\to q}(i,j) > \tau$. As we increase τ from 0 to 1, the true positive rate drops from 1 to 0, whereas false positive rate rises from 0 to 1. The AUC for $\mathbb{C}^{\tau}_{x_{p,i}}$ is then computed as $\sum_{x_{q,k}\in\mathcal{V}} \delta(\mathbf{F}(i,j), \mathbf{F}(i,j))$. Hence, $acc(p \to q)$ is the AUC averaged over the set $\mathcal{W}_{p\to q}$.

4.4 Evaluation

Evaluation is done for every consecutive pair of yearly time intervals. Since we extracted a total of 11 such time intervals, we were able to evaluate the approach for 10 pairs of consecutive time intervals.

We also point out that link prediction is only be performed for nodes which exist in the earlier time interval; link prediction for nodes that do

[1] An ROC curve is a graph of true positive rate (which is, in our case, the fraction of new-adjacencies correctly classified) against false positive rate (which is, in our case, the fraction of non-adjacencies wrongly classified) produced by varying a parameter or threshold of a classifier.

not yet exist is of lesser interest to us. For the relations `Authorship`, `PPublication`, `PCitation` and `SubDomain`, no new adjacencies are formed between existings nodes, i.e. $|\mathcal{U}| = |\mathcal{W}| = 0$. Thus, we do not peform link prediction for these four relations. Instead, link prediction is only performed for the relations `CoAuthorship`, `APublication`, `ACitation`, `Affiliation` and `CommonTopic`. In addition, the relations `APublication`, `ACitation` and `Affliation` are directed relations, and so we also perform link prediction for the reverse direction of these relations. In total, we perform link prediction for eight relations.

For our experiments, we set $\alpha = 0.5$ and tested for $\rho = 0, 0.25, 0.5, 0.75, 1$.

We also evaluated our balanced node similarity measure for the multi-network comprising of all modes and relations, and also for the single-relations networks comprising of only one relation. That is, for the multi-network, we compute a node similarity matrix $\tilde{\mathbf{S}}_p$ for each mode, and then use these similarity matrices for link prediction; for each single-relation network comprising relation $\mathbf{A}_{p\to q}$, we compute the node similarity matrices $\tilde{\mathbf{S}}_p$ and $\tilde{\mathbf{S}}_q$, which are then used for predicting the relation $\mathbf{A}_{p\to q}$ and the reverse direction $\mathbf{A}_{q\to p} = \mathbf{A}_{p\to q}^T$.

4.5 Results

In the interest of space, we will only show the results averaged over the 10 pairs of yearly time intervals. Overall results are presented in Table 2.

We analyze the results in this section, and highlight our key observations in bold. We remind the reader that with $\rho = 0$, the balanced model essentially reduces to a weighted common-neighbors model, while with $\rho = 1$, the balanced model is purely a recursive neighborhood similarity model.

Balanced model with single-relation network performs at baseline. Table 3 shows the improvements in average ranking accuracy that are obtained over the baseline when the balanced model is used. For $\rho = 0, 0.25, 0.5, 0.75$, the difference in average ranking accuracy between the baseline and balanced model is negligible. The low standard deviation shown in Table 4 indicates that the negligible difference is consistently observed. Thus, in the absence of additional information from other relations, our balanced model's performance is neither significantly better or worse than the baseline weighted common neighbors method.

We do note that for $\rho = 1$, the pure recursive neighborhood similarity model is susceptible to worse performance (see `ACitation` and the reverse `ACitation`). This is consistent with the results obtained in Liben-Nowell & Kleinberg [14], where a common-neighbors link predictor outperformed

Table 2: Average Accuracy

ρ	#relations	CoAuthorship	Affiliation	ACitation	APublication
Baseline	One	61.89%	53.11%	69.64%	87.05%
0	One	61.88%	53.11%	69.64%	87.05%
	All	79.11%	57.63%	81.48%	86.09%
0.25	One	61.91%	53.11%	69.68%	87.25%
	All	79.82%	69.84%	82.11%	86.89%
0.5	One	61.91%	53.11%	69.66%	87.49%
	All	80.11%	70.09%	81.95%	87.31%
0.75	One	61.91%	53.11%	69.44%	87.75%
	All	80.12%	70.20%	81.04%	87.65%
1	One	61.88%	53.11%	63.96%	87.78%
	All	71.08%	68.75%	67.99%	87.86%
Maximum		80.12%	70.20%	82.11%	87.86%
ρ	#relations	CommonTopic	Affiliation (reverse)	ACitation (reverse)	APublication (reverse)
Baseline	One	55.74%	52.09%	74.92%	56.85%
0	One	55.74%	52.09%	74.92%	56.85%
	All	65.79%	75.62%	76.05%	62.35%
0.25	One	55.75%	52.09%	74.97%	57.06%
	All	66.66%	77.08%	76.49%	63.40%
0.5	One	55.75%	52.09%	74.98%	57.33%
	All	67.57%	77.30%	76.64%	64.04%
0.75	One	55.75%	52.09%	74.80%	57.69%
	All	68.94%	77.33%	76.56%	64.48%
1	One	55.75%	52.09%	68.25%	57.67%
	All	70.14%	68.80%	68.82%	62.30%
Maximum		70.14%	77.33%	76.64%	64.48%

Table 3: Improvement in Average Accuracy (with Single Relation) Over Baseline

ρ	#relations	CoAuthorship	Affiliation	ACitation	APublication
0	One	0.00%	0.00%	0.00%	-0.01%
0.25	One	0.02%	0.00%	0.04%	0.20%
0.5	One	0.02%	0.00%	0.02%	0.44%
0.75	One	0.02%	0.00%	-0.20%	0.70%
1	One	0.00%	0.00%	-5.68%	0.73%

ρ	#relations	CommonTopic	Affiliation (reverse)	ACitation (reverse)	APublication (reverse)
0	One	0.00%	0.00%	0.00%	0.00%
0.25	One	0.00%	0.00%	0.05%	0.22%
0.5	One	0.00%	0.00%	0.06%	0.49%
0.75	One	0.00%	0.00%	-0.12%	0.84%
1	One	0.00%	0.00%	-6.67%	0.82%

SimRank.

Balanced model with multi-network outperforms baseline model. The improvements that are achieved by the balanced model over the baseline are presented in Table 5. For the balanced models with $\rho = 0, 0.25, 0.75$, significant improvements in average ranking accuracies are observed for five of the eight predicted relations (`Affiliation`, `CoAuthorship`, `ACitation`, `CommonTopic`, reverse `Affiliation`) and small improvements are observed for two other relations (reverse `ACitation`, reverse `APublication`). There is little difference in average ranking accuracy for `APublication`. We also note that models with less extreme values of ρ in generally performed better than those with $\rho = 0, 1$. The low standard deviations in Table 6 indicate that our observations are consistent. (Link prediction on `CommonTopic` has an inherently higher variance due to smaller number of `Journals`.)

Balanced model exploits information from multi-network to improve link predication accuracy. Table 7 shows the improvement in average ranking accuracy when the node similarities are computed using the multi-network, versus using a single-relation network. Results here mirror that in Table 5, showing varying degrees of improvement, from no difference (`APublication`) to large improvements (reverse `Affiliation`). This suggests that (1) information encoded in other relations is useful for link prediction; and (2) our balanced method for computing multi-network link

Table 4: Standard Deviation of Improvement in Accuracy (with Single Relation) Over Baseline

ρ	#relations	CoAuthorship	Affiliation	ACitation	APublication
0	One	0.00%	0.00%	0.00%	0.00%
0.25	One	0.03%	0.00%	0.03%	0.12%
0.5	One	0.03%	0.00%	0.08%	0.21%
0.75	One	0.03%	0.00%	0.26%	0.36%
1	One	0.07%	0.01%	3.29%	0.55%
ρ	#relations	CommonTopic	Affiliation (reverse)	ACitation (reverse)	APublication (reverse)
0	One	0.00%	0.00%	0.00%	0.00%
0.25	One	0.01%	0.00%	0.04%	0.24%
0.5	One	0.01%	0.00%	0.05%	0.55%
0.75	One	0.01%	0.00%	0.15%	0.94%
1	One	0.01%	0.01%	3.63%	2.09%

Table 5: Improvement in Average Accuracy (with Multi-Network) Over Baseline

ρ	#relations	CoAuthorship	Affiliation	ACitation	APublication
0	All	17.23%	4.52%	11.84%	-0.96%
0.25	All	17.93%	16.73%	12.47%	-0.16%
0.5	All	18.23%	16.98%	12.30%	0.25%
0.75	All	18.24%	17.09%	11.40%	0.59%
1	All	9.20%	15.64%	-1.66%	0.81%
ρ	#relations	CommonTopic	Affiliation (reverse)	ACitation (reverse)	APublication (reverse)
0	All	10.05%	23.53%	1.13%	5.50%
0.25	All	10.91%	24.99%	1.57%	6.55%
0.5	All	11.82%	25.21%	1.72%	7.19%
0.75	All	13.19%	25.24%	1.64%	7.63%
1	All	14.40%	16.70%	-6.10%	5.45%

Table 6: Standard Deviation of Improvement in Accuracy (with Multi-Network) Over Baseline

ρ	#relations	CoAuthorship	Affiliation	ACitation	APublication
0	All	1.94%	1.31%	2.19%	0.40%
0.25	All	2.08%	1.93%	2.27%	0.43%
0.5	All	2.15%	1.89%	2.12%	0.35%
0.75	All	2.27%	1.81%	1.93%	0.32%
1	All	3.43%	1.25%	3.87%	0.49%
ρ	#relations	CommonTopic	Affiliation (reverse)	ACitation (reverse)	APublication (reverse)
0	All	8.18%	2.36%	0.79%	3.35%
0.25	All	8.20%	1.80%	1.04%	3.56%
0.5	All	8.39%	1.75%	1.12%	3.64%
0.75	All	9.29%	1.73%	1.23%	3.72%
1	All	9.66%	1.91%	3.89%	3.94%

similarity is able to exploit such information.

Number of iterations to achieve convergence of node similarity computation increases with ρ. This trend is shown in Figure 2. As ρ increases, greater emphasis is placed on similarity from multi-hop relationships versus immediate common neighbors. Thus, with larger values of ρ, more iterations are required to achieve convergence (defined by $(\forall p, i, j)|\tilde{\mathbf{S}}^{(k)}_{p,i,j} - \tilde{\mathbf{S}}^{(k-1)}_{p,i,j}| < 10^{-5}$).

Furthermore, we note that convergence is quickly achieved in less than 10 iterations for $\rho < 1$. In addition, we observed epirically that convergence is always achieved for all values of ρ, for all networks, across all years.

5 Related Work

The problem of link prediction has gathered increasing attention in the past decade. In this section, we discuss some related work and where appropriate, make comparisons with our approach.

5.1 Link prediction using node similarities

Liben-Nowell & Kleinberg [14] presented a survey of graph proximity or "similarity" measures. Each such measure assigned a connection weight

Table 7: Improvement in Average Accuracy with Multi-Network over Single-Relation Network

ρ	#relations	CoAuthorship	Affiliation	ACitation	APublication
0	From single-relation to multi-network	17.23%	4.52%	11.84%	-0.96%
0.25		17.91%	16.73%	12.43%	-0.36%
0.5		18.20%	16.98%	12.28%	-0.19%
0.75		18.21%	17.09%	11.60%	-0.11%
1		9.20%	15.64%	4.03%	0.07%
ρ	#relations	CommonTopic	Affiliation (reverse)	ACitation (reverse)	APublication (reverse)
0	From single-relation to multi-network	10.05%	23.53%	1.13%	5.50%
0.25		10.91%	24.99%	1.52%	6.33%
0.5		11.82%	25.21%	1.65%	6.70%
0.75		13.19%	25.24%	1.76%	6.79%
1		14.39%	16.70%	0.57%	4.63%

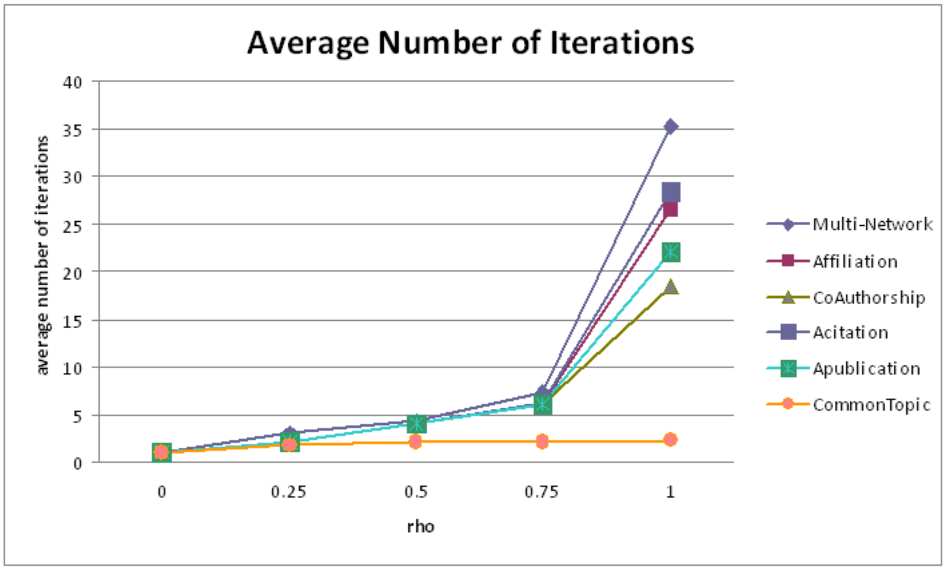


Figure 2: Average number of iterations for convergence of balanced model

score(x, y) to every pair of (homogeneous) nodes (x, y). By making the homophily [12] assumption that nodes that are similar are more likely to associate with each other, the authors are able to generate a ranking of likelihood of adjacency formation based on the pair-wise node similarity values. Unfortunately, the link prediction is restricted to only relations between nodes of the same type. In this report, we adopt the general framework of Liben-Nowell & Kleinberg in applying similarity measures to the task of link prediction. However, we consider our technique as an improvement in two major ways:

- Firstly, while the class of similarity measures in Liben-Nowell & Kleinberg were applied to uni-modal uni-relation co-authorship networks, we demonstrate a multi-network similarity measure in this paper.
- Secondly, our link prediction technique is vastly different from the straightforward similarity-based ranking of Liben-Nowell. We assume link preference (a node is likely to form links with another node which is similar to the nodes that it is already linked with) instead of homophily (a node is likely to form link with another node which is similar to itself). We also point out that the similarity-based ranking is not extensible to the multi-modal setting, whereas our link prediction technique is easily applied to predicting relations across two types of nodes.

Of the similarity measures covered in [14], common neighbors and SimRank [7] bear the greatest resemblance to our method. The intuition behind SimRank is that similar objects are related to similar objects. This resonates with our notion of neighborhood similarity, although there are difference in the exact implementation details (such as the form of normalization). Our similarity measure can thus be seen as a combination of the SimRank and common neighbors models, but further extended to the multi-network setting. It is interesting to note that the link prediction results presented in this paper are consistent with those in Liben-Nowell & Kleinberg [14]: a common neighbors predictor ($\rho = 0$) tends to perform better than one based on neighborhood similarity ($\rho = 1$) or SimRank. Nevertheless, the best results are obtained by the balanced models with $\rho = 0.25, 0.5, 0.75$.

5.2 Statistical Relational Learning

A class of popular approach for modeling relational data is Statistical Relational Learning (SRL). A survey of such methods is provided in Getoor [4, 5]. We provide a brief description of a few SRL models below; for a in-depth treatment of the topic, please refer to Getoor & Taskar [6].

Some of the earlier works in this field were by Popescul et. al. [16, 15, 17], who proposed Structural Logistic Regression for link analysis. Structural

Logistic Regression couples two main processes: (1) generation of features from relational data and (2) their selection with statistical model selection criteria. Thus, Structural Logistic Regression combines concepts from inductive logic programming (for generation of features) and machine learning (for selection of features). The two processes are executed iteratively, so that generated features are selected by a statistical model selection, and in turn selected features are used to generate new features from the relational data. In [16, 15, 17], the binary logistic regression model was the statistical model of choice, although the authors point out that it may be possible to use other multi-class statistical classifiers as well.

Other SRL approaches like Probabilistic Relational Models (PRM) [3], Relational Markov Networks (RMN) [23], etc. define probabilistic models over the relational data. A PRM is in essence a directed probabilistic graphical model, where a random variable in the PRM corresponds to an attribute of an entity or potential adjacency in the network. Reference and existence of potential adjacencies can be modeled as attributes as well. The probability distribution over a random variable is then dependent on the other attributes of the entity or adjacency, and possibly on the attributes of related entities and adjacencies too. Importantly, the parameters of probability distributions are shared between attributes of the same type.

RMNs are the undirected analogy to PRMs. Cliques are induced on the set of entities/adjacencies and their attributes by the use of *clique templates*. Each clique template performs a kind of SQL-style query on the entities/adjacencies by selecting the appropriate attributes of entities which are related in the specified way. Parameter sharing between cliques is acheived by defining potential on clique templates rather than individual cliques.

Domingos & Richardson [2] describe *Markov logic*, a unifying framework for SRL methods that combines undirected probabilistic graphical models (Markov networks) and first-order logic. Syntactically, Markov logic augments first-order logic with a weight for every formula. Semantically, a set of Markov logic formulae represents a probability distribution over possible worlds, in the form of a log-linear model with one feature per grounding of a formula in the set, with the corresponding weight. Said differently, given a set of constants representing the entities in the world, a Markov network is then induced, such that cliques in the Markov network correspond to the Markov logic formulas, with log-linear potentials with the corresponding weights. Domingos & Richardson also show how other SRL approaches, including Structural Logistic Regression, PRMs and RMNs, map into Markov logic models.

More recently, Xu et. al. proposed the use of Infinite Hidden Relational models (IHRM) [29] and Multi-Relational Gaussian Processes (MRGP) [28] specifically for multi-relational learning. An IHRM introduces a random variable for each potential link, and also an additional *hidden* random variable for every entity. The hidden random variable can be seen as a hidden attribute specifying the cluster to which the entity belongs, and is assume to determine the attributes of the entity. Links are also assumed to depend only on the hidden random variables of the two entities involved. The number of clusters is allowed to be infinitely large by using a Dirichlet process mixture model.

Like IHRMs, MRGPs also have a latent variable for each entity, encoding the essential property of the entity. In additional, another latent variable is introduced for each entity and relation that it can be involved in, representing the hidden causes for the entity to be involved in the relation. The MRGP differs from the IHRM in that the latent/hidden variables are outputs from Gaussian processes. The likelihood of a link formation is then dependent on the essential properties and hidden causes of the entities involved in the relation.

Our proposed solution for multi-network link prediction clearly differs from such statistical modeling of relational data. We do not deny the expressive power of SRL models, but nevertheless point out some its potential difficulties and shortcomings in link prediction. As noted in [4], the typically small prior probability of a link causes difficulty for building statistical models for link prediction. More importantly, exact inference using probabilistic models tends to be computationally expensive, and in most situations only approximate inference is possible.

Furthermore, the typical problem setting in the SRL papers differs from our temporal link prediction setting. In the SRL papers, it is often assumed that links are only partially observed; the problem is then to resolve the uncertainty of unobserved potential links. On the other hand, in this report, our interest lies in predicting formation of potential links at a later time $\mathbb{t}+1$, given that we have observed links at an earlier time $\mathbb{t}$. While we imagine that it is possible to extend SRL models to incorporate a temporal aspect, the temporal link prediction problem is nonetheless not pursued in the SRL papers. Conversely, we do not pursue the problem of unobserved links in this report.

5.3 Matrix factorization

Another class of approaches to multi-relational modeling is matrix factorization. In these approaches, the adjacency or attribute matrices are decomposed into low rank factors. It is necessary to determine beforehand the ranks (or numbers of clusters) of the factorization. In [11], Long et. al. propose a Collective Factorization on Related Matrices for the purpose of clustering. Specifically, to cluster each mode $\mathcal{X}_p$ into c_p clusters, the form of factorization employed is $\mathbf{A}_{p\to q} \approx \mathbf{C}_p\mathbf{M}_{p\to q}\mathbf{C}_q^T$, where $\mathbf{C}_p \in \{0,1\}^{n_p\times c_p}$ is the cluster indicator matrix such that $\sum_{i=1}^{c_p} \mathbf{C}_{p\to q}(i,j) = 1$ and $\mathbf{C}_{p\to q}(i,j) = 1$ denotes that $x_{p,i}$ is associated with the jth cluster. $\mathbf{M}_{p\to q}$ is the cluster association matrix such that $\mathbf{M}_{p\to q}(i,j)$ denotes the association between the ith cluster of mode $\mathcal{X}_p$ and the jth cluster of $\mathcal{X}_q$. An approximate factorization is then achieved by minimizing a loss function comprised of the Frobenius norms of $\mathbf{A}_{p\to q} - \mathbf{C}_p\mathbf{M}_{p\to q}\mathbf{C}_q^T$ of every relation.

Tang et. al. [22, 21] then extend this model by including a temporal aspect, and imposing additional loss from changes in the cluster membership over time. The temporal collective factorization model is then applied to community detection in dynamic multi-mode networks.

Singh & Gordon [19] suggest an alternative form of factorization: $\mathbf{A}_{p\to q} \approx f(\mathbf{L}_p\mathbf{L}_q^T)$, where $\mathbf{L}_p \in \mathbb{R}^{n_p\times c}$ is the low rank factors for mode $\mathcal{X}_p$, $f : \mathbb{R}^{n_p\times n_q} \mapsto \mathbb{R}^{n_p\times n_q}$, and c is the rank of factorization. Singh & Gordon propose an iterative Newton-Rapshon solution based on minimizing the Bregman divergences between the model and the relation matrices. Lippert et. al. [10] adopt a similar factorization as Singh & Gordon, but minimize an alternative objective through gradient descent. Both Singh & Gordon and Lippert et. al. apply their collective matrix factorizations to the task of predicting unobserved links.

5.4 Other domains

To the best of our knowledge, Li et. al. [9] come closest to our approach of modeling multi-network node similarities. Li et. al. are primarily concerned with the different problem of content-based image retrieval (CBIR), and to that end, propose four inter-dependent similarity matrices: $\mathbf{S_B}$ for blobs, $\mathbf{S_W}$ for words, $\mathbf{S_{T_B}}$ for images calculated on their constituent blobs, and $\mathbf{S_{T_W}}$, for images calculated on their constituent words. A iterative estimation of the similarity matrices, similar to our approach, is proposed. In the proposed types of iteractions between similarity matrices, each similarity matrix is influenced by at most one other similarity matrix.

6 Conclusion and Future Directions

We have presented a method for computing multi-network node similarities, and discussed how link prediction can be performed by adjacency propagation through the similarity matrices. Our experiment demonstrates that (1) information encoded in other relations is useful for link prediction; and (2) our balanced method for computing multi-network link similarity is able to exploit such information, improving link prediction by up to 25%.

In this report, we have validated our model on one dataset, albeit on multiple different types of relations. It would be interesting to apply the model to other datasets of different nature, and to analyse any differences in performance.

Furthermore, the node similarity values may potentially be applied to network analysis problems other than link prediction. For instance, Multidimensional scaling may be applied to the node similarity matrices for clustering and community derivation.

We have thus far been focusing on exploiting structural information for node similarity estimation and link prediction. It may be possible to further improve link prediction accuracy by incorportating other non-topological information, e.g. entity attributes, into the computation of node similarities.

Finally, other possible future directions of research are to explore using different parameters $\rho_{p \to q}$ for each relation instead of a single parameter ρ, and to automatically learn the optimal ρ or $\rho_{p \to q}$ values from data.

References

[1] Fan R. K. Chung. *Spectral Graph Theory (CBMS Regional Conference Series in Mathematics, No. 92)*. American Mathematical Society, February 1997.

[2] Pedro Domingos and Matthew Richardson. Markov logic: A unifying framework for statistical relational learning. In *Proceedings of The ICML-2004 Workshop on Statistical Relational Learning and its Connections to Other Fields*, pages 49–54, 2004.

[3] Lisa Getoor, Nir Friedman, Daphne Koller, and Benjamin Taskar. Learning probabilistic models of link structure. *J. Mach. Learn. Res.*, 3:679–707, 2003.

[4] Lise Getoor. Link mining: a new data mining challenge. *SIGKDD Explor. Newsl.*, 5(1):84–89, July 2003.

[5] Lise Getoor and Christopher P. Diehl. Link mining: a survey. *SIGKDD Explor. Newsl.*, 7(2):3–12, December 2005.

[6] Lise Getoor and Ben Taskar. *Introduction to Statistical Relational Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2007.

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

[8] D. Jensen. Dataset: Hep-th, 2003.

[9] M. Li, X. B. Xue, and Z. H. Zhou. Exploiting multi-modal interactions: A unified framework. In *Proc. IJCAI '2009*, pages 1120–1125, Pasadena, CA, July 2009.

[10] Christoph Lippert, Stefen H. Weber, Yi Huang, Volker Tresp, Matthias Schubert, and Hans-Peter Kriegel. Relation prediction in multi-relational domains using matrix factorization. In *NIPS 2008 Workshop on Structured Input Structure Output*, 2008.

[11] Bo Long, Zhongfei . Zhang, Xiaoyun Wú, and Philip S. Yu. Spectral clustering for multi-type relational data. In *Proceedings of the 23rd international conference on Machine learning*, pages 585–592, Pittsburgh, Pennsylvania, 2006. ACM.

[12] Miller McPherson, Lynn S. Lovin, and James M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444, 2001.

[13] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Neural Information Processing Systems 14*, pages 849–856. MIT Press, 2001.

[14] David L. Nowell and Jon Kleinberg. The link prediction problem for social networks. In *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management*, pages 556–559, New York, NY, USA, 2003. ACM.

[15] A. Popescul, L. Ungar, S. Lawrence, and D. Pennock. Statistical relational learning for document mining. 2003.

[16] Alexandrin Popescul, Rin Popescul, and Lyle H. Ungar. Statistical relational learning for link prediction. 2003.

[17] Alexandrin Popescul and Lyle H. Ungar. Structural logistic regression for link analysis. In S. Džeroski, Luc De Raedt, and Stefan Wrobel, editors, *Proceedings of the 2nd International Workshop on Multi-Relational Data Mining (MRDM-2003)*, pages 92–106, New York, 2003. ACM.

[18] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[19] Ajit P. Singh and Geoffrey J. Gordon. Relational learning via collective matrix factorization. In *KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 650–658, New York, NY, USA, 2008. ACM.

[20] Volker Strassen. Gaussian elimination is not optimal. *Numer. Math.*, 13:354–356, 1969.

[21] L. Tang, H. Liu, and J. Zhang. Identifying evolving groups in dynamic multi-mode networks. *IEEE Trans. Knowl. and Data Eng.*, to appear:to appear, to appear 2010.

[22] Lei Tang, Huan Liu, Jianping Zhang, and Zohreh Nazeri. Community evolution in dynamic multi-mode networks. In *KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 677–685, New York, NY, USA, 2008. ACM.

[23] Ben Taskar, Ming F. Wong, Pieter Abbeel, and Daphne Koller. Link prediction in relational data. In *in Neural Information Processing Systems*, 2003.

[24] L. N. Teow and D. Katabi. Iterative collaborative ranking of customers and providers. Technical Report MIT-CSAIL-TR-2006-050, CSAIL, Massachusetts Institute of Technology, Cambridge, MA, July 2006.

[25] A. K. C. S. Vanderbilt and G. Strauss. Custom ontologies for expanded network analysis. In *RTO-MP-IST-063 Visualising Network Information*, pages 6–1 – 6–10.

[26] Fei Wang and Changshui Zhang. Label propagation through linear neighborhoods. In *ICML 2006*.

[27] S. Wasserman and K. Faust. *Social Network Analysis: methods and applications*. Cambridge University Press, 1994.

[28] Zhao Xu, Kristian Kersting, and Volker Tresp. Multi-relational learning with gaussian processes. In *IJCAI'09: Proceedings of the 21st international jont conference on Artifical intelligence*, pages 1309–1314, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.

[29] Zhao Xu, Volker Tresp, Shipeng Yu, and Kai Yu. Nonparametric relational learning for social network analysis. In *Proceedings of the 2nd SNA-KDD Workshop '08*, August 2008.

[30] Dengyong Zhou, Olivier Bousquet, Thomas N. Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *Advances in Neural Information Processing Systems 16*, 2004.

Part B: Interaction Behavior

Mining Interaction Behaviors from Information Exchange Networks

Mining Interaction Behaviors from Information Exchange Networks

Lim, Ee-Peng
School of Information Systems
Singapore Management University
eplim@smu.edu.sg

1 Overview

In this project, we investigate characterization and measurement of interaction behaviors in information exchange networks based on user-generated interaction data. We will focus on multimodal information exchange networks which involve actors sending information to one another. Examples of such networks include email, messaging, and blog networks.

We focus on modeling engagingness[1] and responsiveness behaviors in email networks and messaging networks. We have used Enron Email data and MyGamma Social Network Message data as the target datasets. The former is so far the only known publicly available information exchange data with messages assigned with specific senders and recipients. Email data preprocessing and thread assembly were conducted on the dataset. We also introduced several engagingness and responsiveness models, and proposed to use them as features in solving the email reply order prediction task.

The MyGamma social network message dataset is from a proprietary mobile social networking site known as myGamma. MyGamma is owned by BuzzCity Pte Ltd, a Singapore company. This dataset offers both messaging and friendship network data for our research. We have adapted our proposed behavior models and developed new ones for this myGamma dataset.

[1] In our previous documents, the term “activeness” was used. Subsequently, we adopt “engagingness” as a more appropriate term.

2 Behavior Modeling in Enron Email Network

The following summarizes the important research contributions of our work in behavior modeling for email networks:

- We define four categories of models for engagingness and responsiveness behaviors prevalent in email networks. They are (a) email based, (b) email thread based, (c) email sequence based, and (d) social cognitive model categories. For each model category, one can define different behavior models based on different email attributes. To the best of our knowledge, this is the first time engagingness and responsiveness behavior models are studied systematically.
- We apply our proposed behavior models on the Enron email network, analyze and compare the proposed behavioral models. We conduct data preprocessing on the email data and establish links between emails and their replies. In our empirical study, we found engagingness and responsiveness are distinct from each other. Most engagingness (responsiveness) models of users are shown to be consistent with each other.
- We introduce email reply order prediction as a novel task that uses engagingness, responsiveness and other email features as input features. An SVM classifier is then learnt from the features of training email pairs and applied to test email pairs. According to our experimental results, the accuracy of our SVM classifier is about 77% which is 50% better than random guess. This indicates that user behaviors are useful in the prediction task.

2.1 Engagingness and Responsiveness Behavior Models

In this section, we describe our proposed behavior models for user engagingness and responsiveness. All the models assume that emails have been preprocessed with duplicate elimination and email reply relationship identification. We divide our models into the following categories:

- **Email based models**: These models consider emails as the basic data units for measuring user behaviors. Email attributes such as sender, recipient list, date, etc., are used.
- **Email thread based models**: These models consider email threads as the basic data units for measuring user behaviors. The models therefore use attributes of email thread to quantify behaviors.

Table 1: Notations.

$S(u_i)$	Emails sent by user u_i
$R(u_i)$	Emails received by u_i
$RB(u_i)$	Email replies sent by u_i
$RT(u_i)$	Emails replying to u_i's earlier emails
$TH(u_i)$	Threads started by an email sent by u_i
$r(e)$	Reply to email e
$Sdr(e)$	Sender of email e
$Rcp(e)$	Recipients (in both To and Cc lists) of email e
$t(e)$	Sent time of email e
$E(u_i \rightarrow u_j)$	Emails from u_i to u_j
$E(u_i \leftrightarrow u_j)$	Emails between u_i and u_j
$rt(u_i \rightarrow u_j)$	Avg. response time from u_i to u_j
$rt(u_i \leftrightarrow u_j)$	Avg. response time between u_i and u_j
$RE(u_i \rightarrow u_j)$	Reply emails from u_i to u_j
$RE(u_i \leftrightarrow u_j)$	Reply emails between u_i and u_j

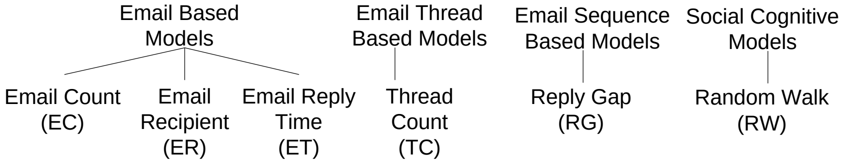


Figure 1: Taxonomy of Models

- **Email sequence based models**: These models examine the sequence of emails received and replied by each user and derive the user behaviors from the gaps between emails received and their replies.

- **Social cognitive models**: These models consider social perception of user behaviors within the email network and measure behaviors accordingly.

Figure 1 shows the taxonomy of behavior models in the above categories to be further defined in the following sections. Each model (M) consists of a pair of engagingness (A^M) and responsive (R^M) score formulas defined based on some principles. The A^M and R^M score values are in [0,1] range with 0 and 1 representing the lowest and highest values respectively. Table 1 shows a list of symbols and their meanings that we use in this report.

2.1.1 Email Based Models

Email Count Model (EC)

The email count model is defined based on the principle that an engaging user should have most of his/her emails replied, while a responsive user should have most of his/her received emails replied. The engagingness and responsiveness formulas are thus defined by:

$$A^{EC}(u_i) = \frac{|RT(u_i)|}{|S(u_i)|} \tag{1}$$

$$R^{EC}(u_i) = \frac{|RB(u_i)|}{|R(u_i)|} \tag{2}$$

For users with empty $S(u_i)$ (or $R(u_i)$), $A^{EC}(u_i)$ (or $R^{EC}(u_i)$) is assigned a zero value.

Email Recipient Model (ER)

The intuition of this model is that an email with many recipients is likely to expect very few replies. Hence, an engaging user is one who gets replies from many recipients of his/her emails while an non-engaging user receives very few or no reply even when his/her emails are sent to many recipients. On the other hand, a responsive user is one who replies emails regardless of the number of recipients in the emails. A non-responsive user is one who does not reply even if the emails are directed to him/her only. The engagingness and responsiveness formulas are thus defined by:

$$A^{ER}(u_i) = \frac{1}{|S(u_i)|} \sum_{e \in S(u_i)} \frac{|\{u_j \in Rcp(e) \wedge r(e) \in RB(u_j)\}|}{|Rcp(e)|} \tag{3}$$

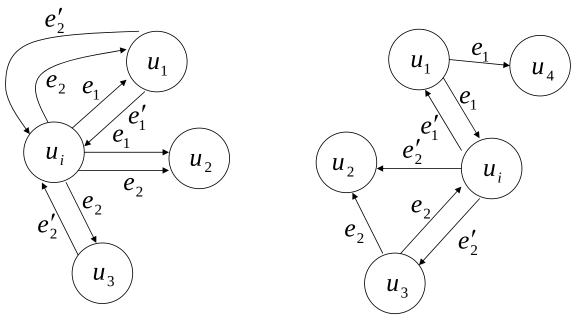
$$R^{ER}(u_i) = \frac{1}{|R(u_i)|} \sum_{\substack{e \in RB(u_i)\ s.t.\\ \exists u_j, \exists e'' \in S(u_j), r(e'')=e}} \frac{|Rcp(e)|}{MaxRcpCnt} \tag{4}$$

where $MaxRcpCnt$ denotes the largest recipient count among all Enron emails.

Email Reply Time Model (ET)

The reply time of an email can be an indicator of user engagingness and responsiveness. The email reply time model adopts the principle that engaging users receives the reply emails sooner than non-engaging users, while responsive users reply to the received emails quicker than non-responsive users.

Given an email e' which is a reply of email e, $e' = r(e)$, the *reply time* of e', $RT(e') = t(e') - t(e)$. The z-normalized reply time $\hat{RT}(e')$ is defined by



(a) engagingness of u_i (b) Responsiveness of u_i

Figure 2: Examples

$\frac{RT(e')-\overline{RT}}{\sigma_{RT}}$ where $\overline{RT}$ and σ_{RT} are the mean and standard deviation of reply time respectively. Now, we define the engagingness and responsiveness of ET model as:

$$A^{ET}(u_i) = \frac{1}{|S(u_i)|} \sum_{e \in S(u_i)} \frac{1}{|Rcp(e)|} \sum_{\substack{u_j \in Rcp(e), \\ \exists e' \in RB(u_j), e'=r(e)}} f(\hat{RT}(e')) \tag{5}$$

$$R^{ET}(u_i) = \frac{1}{|R(u_i)|} \sum_{e' \in RB(u_i), e \in R(u_i), r(e)=e'} f(\hat{RT}(e')) \tag{6}$$

where

$$f(x) = \frac{e^{-x}}{1+e^{-x}} \tag{7}$$

The function $f()$ is designed to convert the normalized reply time to the range [0,1] with 0 and 1 representing extreme slow and extreme fast reply times respectively.

Examples

Consider the email network in Figure 2(a). Suppose e'_k denote the reply to email e_k. The engagingness values of u_i derived by the EC and ER email based models are: (a) $A^{EC} = \frac{3}{5} = 0.6$; and (b) $A^{ER} = \frac{\{\frac{1}{2}+\frac{2}{3}\}}{2} = 0.58$. Suppose $\hat{RT}(e'_1) = 5$, $\hat{RT}(e'_2) = 10$, and $\hat{RT}(e'_3) = 20$. The engagingness of u_i according to ER model is:

$$A^{ET} = \frac{\frac{1}{2} \cdot (f(5)) + \frac{2}{3} \cdot (f(10) + f(20))}{2} = 0.45$$

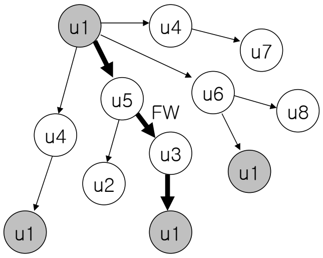


Figure 3: Email thread example.

Table 2: Distribution of emails per thread.

# emails	2	3	4	5	6	≥ 7	Total
# threads	11,302	3,925	1,614	732	404	616	18,593

Consider the email network in Figure 2(b). The responsiveness values of u_i derived by EC and ER models are: (a) $R^{EC} = \frac{2}{2} = 1$; and (b) $R^{ER} = \frac{\{\frac{1}{4}+\frac{2}{4}\}}{2} = 0.38$.

2.1.2 Email Thread Based Model

Here, we define the **thread count model (TC)** as an email thread based model. In the email count model, engagingness is measured by emails sent by a sender and sent emails directly replied by some recipient(s). However, direct reply is not the only type of response to an email. Email may be indirectly replied in email threads due to forwarded emails. For example, as illustrated in Figure 3, a user u_1 advertises a job position by sending an email to u_5 who subsequently forwards the email to his student u_3. If u_3 replies to u_1, we say that the original email is replied indirectly in an email thread.

Email thread is defined by a tree of emails connected by reply and forward relationships. Table 2 shows the distribution of threads by the number of emails per thread. As we can notice, the distribution follows Zipf's law. Majority of threads (11,302) contain only two emails. There are 3925 threads that include three emails. The largest thread contains 37 emails.

Based on email threads, the thread count model includes indirect replies to emails forwarded between users using the principle: the user is highly engaging if he or she receives many of his/her emails replied directly or indirectly by recipients, and is highly responsive if he or she replies or forwards most emails earlier received. In the following, the engagingness and responsiveness of a user u_i are defined as:

$$A^{TC}(u_i) = \frac{1}{|S(u_i)|} \cdot |\{e \in S(u_i) | \exists t \in TH(u_i), \exists e', e \underset{t}{\twoheadrightarrow} e' \wedge u_i \in Rcp(e')\}| \tag{8}$$

$$R^{TC}(u_i) = \frac{1}{|R(u_i)|} \cdot |\{e \in R(u_i) | \exists u_j, e', t \in TH(u_j), e \underset{t}{\twoheadrightarrow} e' \wedge u_j \in Rcp(e')\}| \tag{9}$$

where $e \twoheadrightarrow_t e'$ returns TRUE when e is directly or indirectly connected to e' in the thread t, and FALSE otherwise.

2.1.3 Email Sequence Based Model

Email sequence refers to the sequence of emails sent and received by a user ordered by time. To derive engagingness and responsiveness from email sequences, we consider the principle that an engaging user is expected to have his or her sent emails replied soon after they are received by the email recipients, and an responsive user replies soon after they receive emails. As users may not always stay online, the time taken to reply an email may vary very much. Instead, we consider the number of emails received later than an email e but are replied before e by a user as a proxy of how soon e is replied.

The above principle is thus used to develop the **reply gap model (RG)**. Let seq_i denote the email sequence of user u_i. When an email received by u_i is replied before other email(s) received earlier, the reply of the former is known as an *out-of-order reply*. Formally, for an email e received by u_i, we define the *number of emails received* and *number of out-of-order replies* between e and its reply e' in seq_i, denoted by $n_r(u_i, e)$ and $n_{\overline{o}}(u_i, e)$ respectively, as

$$n_r(u_i, e) = \begin{cases} \text{\# emails received between} & \text{if } \exists e' \in RT(u_i), \\ e \text{ and } e' \text{ in } seq_i, & \quad r(e) = e' \\ -1, & \text{otherwise} \end{cases} \tag{10}$$

$$n_{\overline{o}}(u_i, e) = \begin{cases} \text{\# emails received} & \text{if } \exists e' \in RT(u_i), \\ \text{between } e \text{ and } e' \text{ in } seq_i & \quad r(e) = e' \\ \text{and have been replied,} & \\ -1, & \text{otherwise} \end{cases} \tag{11}$$

The -1 value is assigned to n_r and $n_{\overline{o}}$ when e is not replied at all. The user engagingness and responsiveness of the RG model are thus defined as:

$$A^{RG}(u_i) = \frac{\sum_{e \in S(u_i)} \left(\frac{1}{|Rcp(e)|} \sum_{u_j \in Rcp(e)} (1 - \frac{n_{\overline{o}}(u_j,e)}{n_r(u_j,e)}) \right)}{|S(u_i)|} \tag{12}$$

$$R^{RG}(u_i) = \frac{\sum_{e \in R(u_i)} (1 - \frac{n_{\overline{o}}(u_i,e)}{n_r(u_i,e)})}{|R(u_i)|} \tag{13}$$

For example, let $seq_i = \{e_1, e_2, e_3, e'_1, e_4, e'_4, e'_2\}$ be the email sequence of user u_i where $e'_k = r(e_k)$'s. Note that $\frac{n_{\overline{o}}(u_i,e_1)}{n_r(u_i,e_1)}$, $\frac{n_{\overline{o}}(u_i,e_2)}{n_r(u_i,e_2)}$, $\frac{n_{\overline{o}}(u_i,e_3)}{n_r(u_i,e_3)}$, and $\frac{n_{\overline{o}}(u_i,e_4)}{n_r(u_i,e_4)}$ are $\frac{0}{3}$, $\frac{1}{2}$, $\frac{1}{1}$, and 0 respectively. Hence, $A^{RG}(u_i) = \frac{1+\frac{1}{2}+0+1}{4} = 0.625$. The responsiveness of u_i can be computed in the same manner.

2.1.4 Social Cognitive Model

A social cognitive model is based on *social cognitive theory* which suggests that people learn by watching what others do [8]. Such kind of models thus measure a user's engagingness and responsiveness behaviors by observing what the other users react to emails sent from the user and observe the email interaction among one another. In this paper, we introduce a **random walk (RW)** social cognitive model.

For engagingness, each user u_k perceives a user u_i to be more engaging than another user u_j if more emails from u_i are replied ahead of emails from u_j based on the emails in the mailbox of u_k. For instance, suppose that u_k has an email sequence $seq_k = \langle e_1(u_1, \{u_k\}), e_2(u_2, \{u_k\}), e'_2(u_k, \{u_2\}), e'_1(u_k, \{u_1\}) \rangle$, where $e_v(u_x, U_y)$ denotes email e_v sent by u_x to recipients U_y and e'_v denotes the reply of email e_v. u_k receives e_1 before e_2 but the reply e'_1 comes after e'_2. This indicates that u_k considers u_2 more important than u_1. Furthermore, u_2 is more engaging than u_1 from u_k's standpoint. Based on the above observation, we say that u_k observes the engagingness superiority of u_2 over u_1.

Similarly for responsiveness, u_k perceives a user u_1 to be more responsive than another user u_2 if u_k observes reply emails from u_1 earlier than u_2 for the same emails sent to both u_1 and u_2 which can be from u_k or other users.

Formally, we represent an **engagingness weighted directed graph** $G^A = \langle U, E^A \rangle$ as follows:

- U represents the set of all users.

- E^A consists of directed edges. When in the mailbox of some u_k, u_i has x_k emails replied ahead of emails from u_j, we represent this by a directed edge $u_j \rightarrow u_i$.

- The weight of $u_j \rightarrow u_i$, $weight(u_j \rightarrow u_i)$, is the sum of x_k's for all u_k's. The larger is $weight(u_j \rightarrow u_i)$, the more users observe that u_i is more engaging than u_j.

In a similar manner, we can define a **responsiveness weighted directed graph** $G^R = \langle U, E^R \rangle$.

The engagingness (or responsiveness) weighted directed graph will be further processed to derive the degree of engagingness (or responsiveness) of users. Each directed graph so far captures the perceived relative difference between users in engagingness (or responsiveness). It however does not immediately assign engagingness/responsiveness scores to the users. We therefore propose to perform random walk on the engagingness (or responsiveness) graph so as to determine the user engagingness (or responsiveness) values as the stationary probabilities of visiting them.

The random walk process on the engagingness graph to obtain the engagingness of users denoted by $A^{RW}(u_k)$'s consists of the following steps:

1. Determine the largest node aggregated edge weight, $MaxWeight = Max_{u_j}\{\sum_{u_i} weight(u_j \rightarrow u_i)\}$

2. For each user u_j,

   (a) $sum_j = 0$

   (b) For each edge $u_j \rightarrow u_i$,

      i. Assign a transition probability to $u_j \rightarrow u_i$ as $p(u_j, u_i) = \frac{weight(u_j \rightarrow u_i)}{MaxWeight}$

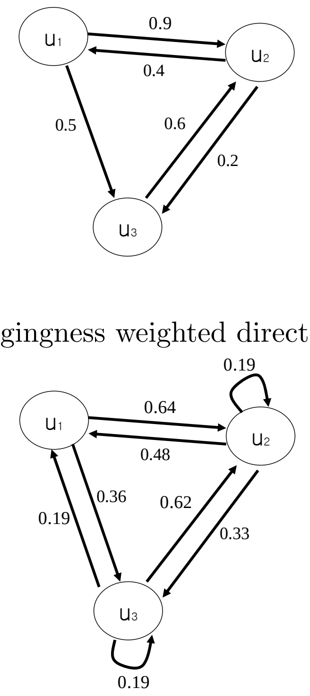
      ii. $sum_j = sum_j + p(u_j, u_i)$

   (c) // assign to the remaining weights to all users.
   Create an edge $u_j \rightarrow u_t$ for all u_t with $p(u_j, u_t) = \frac{1-sum_j}{|U|}$ if $u_j \rightarrow u_t$ does not exist;
   Assign $p(u_j, u_t) += \frac{1-sum_j}{|U|}$ otherwise

3. For each user u_i, initialize $A^{RW}_{new}(u_i)$ randomly

4. Repeat the following steps:

   (a) For each u_i, $A^{RW}(u_i) = A^{RW}_{new}(u_i)$



(a) engagingness weighted directed graph

(b) engagingness graph for random walk

Figure 4: Social cognitive model.

(b) For each u_i, $A^{RW}_{new}(u_i) = \sum_{u_j \to u_i} p(u_j, u_i) \cdot A^{RW}(u_j)$

5. Until $|A^{RW}(u_i) - A^{RW}_{new}(u_i)| \leq \epsilon$ [2] for all u_i's

To illustrate the above algorithm, consider the example in Figure 4. u_2 is more engaging than u_1, with $weight(u_1 \to u_2) = 0.9$. On the other hand, u_1 is more engaging than u_2 with $weight(u_2 \to u_1) = 0.4$. In Figure 4 (a), the total engagingness weight of u_1 to all nodes u_2 and u_3 in the engagingness weighted directed graph is $weight(u_i) = weight(u_1 \to u_2) + weight(u_1 \to u_3) = 1.4$. In the same way, the engagingness weight of u_2 and u_3 are 0.6 and 0.6, respectively. Then, the weight value of each link is normalized by the maximum weight value, $MaxW = weight(u_1)$. E.g., $weight(u_2 \to u_3) = \frac{weight(u_2 \to u_3)}{MaxW} = \frac{0.2}{1.4}$. For nodes with total weight < 1, the unused weight will be used to create links with equal weights to all the nodes. E.g., for u_2, it has unused weight of $\frac{\{MaxW - weight(u_2)\}}{weight(u_1)} = \frac{\{1.4-0.6\}}{1.4}$. As a result of the new links for the unused weight, $weight(u_2 \to u_3) = \frac{0.2}{1.4} + \frac{\{1.4-0.6\}}{1.4} \cdot \frac{1}{3}$=0.33. In this process, the engagingness graph is row-stochastic because its rows are nonnegative and the sum of each row is one. This stochastic matrix can be viewed as a transition matrix associated to a family of Markov chains, where each entry (u_i,u_j) represents the probability of a transition from state u_i to state u_j.

2.2 Email Reply Order Prediction

We now consider the email reply order prediction which has the following setup. Given a pair of emails (e_i, e_j) sent to the same user from users u_i and u_j respectively, we want to determine the order in which the two emails will be replied. Here, we assume that both e_i and e_j require some replies and u_i and u_j are not the same person. The outcome of prediction is either e_i or e_j first.

Our proposed method is to train a Support Vector Machine (SVM) classifier using labeled email pairs, and to apply the trained classifier on unseen email pairs. For each email pair, we can derive features directly from the emails themselves and their senders including the previous emails they have sent and received. There are three types of features used, namely: (a) *comparative email features* ($\mathbb{E}$), (b) *comparative interaction features* ($\mathbb{I}$) and (c) *comparative behavior features* ($\mathbb{B}$).

[2] In our experiment, we used $\epsilon = .0000001$ and numbers of iterations required to compute A^{RW} and R^{RW} are 8 and 12 respectively.

Table 3: Email Features $\mathbb{E}$.

No	Description	No	Description
1	$t(e)$	9	$\|S(Sdr(e))\|$
2	$size(e)$	10	$\|R(Sdr(e))\|$
3	$size(r(e))$ (assuming we can determine the reply)	11	Avg. $\|S(Sdr(e))\|$ per day
		12	Avg. $\|R(Sdr(e))\|$ per day
4	$size(e) + size(r(e))$	13	$\frac{\|RB(Sdr(e))\|}{\|S(Sdr(e))\|}$
5	$Rcp(e)$	14	$\frac{\|RT(Sdr(e))\|}{\|R(Sdr(e))\|}$
6	$indegee(Sdr(e))$ (# users sending emails to $Sdr(e)$)	15	$\frac{\|RT(Sdr(e))\|}{\|S(Sdr(e))\|}$
		16	$\frac{\|RB(Sdr(e))\|}{\|R(Sdr(e))\|}$
7	$outdegee(Sdr(e))$ (# users receiving emails from $Sdr(e)$)	17	Avg response time for emails in $RT(Sdr(e))$
8	$indegree(Sdr(e))+$ $outdegree(Sdr(e))$	18	Avg response time for emails in $RB(Sdr(e))$

Table 3 lists the email features used in our classifier. For each email feature f_k, we derive a corresponding comparative feature f_k^c of an email pair (e_i, e_j) by $[(e_i, e_j).f_k^c = e_i.f_k - e_j.f_k$. For email send time $t(e)$ feature, we further convert the positive and negative comparative feature values to 1 and -1 respectively. Interaction features refer to set of features derived from the sender of the email to the common recipient u_r as shown in Table 4. The behavior features refer to the six A^M and six R^M behavior scores of email senders. The comparative interaction and behavior features are defined similar to that of email features.

2.3 Experiments - Analysis and Comparison of Behavior Models

The first set of experiments is to evaluate and compare the four types of behavior models on Enron dataset. To compare the ranked user lists produced by two models, we utilize the **Kendall τ distance measure**. In each ranked list, first and last ranked users represent the most and least engaging (or responsive) users respectively. Formally, we denote the rank of a user u_i in a ranked list L_k by $l_k(u_i)$. The Kendall τ distance between two ranked lists L_1 and L_2 is defined as $\frac{K(L_1,L_2)}{\frac{1}{2}n(n-1)}$ such that $K(L_1, L_2) = |(u_i, u_j) : u_i < u_j, (l_1(u_i) < l_1(u_j) \wedge l_2(u_i) > l_2(u_j)) \vee (l_1(u_i) > l_1(u_j) \wedge l_2(u_i) < l_2(u_j))|$. Note that Kendall τ distance is 0 if $l_1 = l_2$ for all users, and 1 if there is no

Table 4: Interaction Features $\mathbb{I}$.

No	Description	No	Description
19	$\lvert E(Sdr(e) \rightarrow u_r)\rvert$	27	$\frac{\lvert RE(Sdr(e) \leftrightarrow u_r)\rvert}{\lvert E(u_r \leftrightarrow Sdr(e))\rvert}$
20	$\lvert E(u_r \rightarrow (Sdr(e))\rvert$	28	$rt((Sdr(e) \rightarrow u_r)$
21	$\lvert E((Sdr(e) \leftrightarrow u_r)\rvert$	29	$rt(u_r \rightarrow (Sdr(e))$
22	$\lvert RE((Sdr(e) \rightarrow u_r)\rvert$	30	# threads involving $(Sdr(e)$, u_j as senders/recipients
23	$\lvert RE(u_r \rightarrow (Sdr(e))\rvert$		
24	$\lvert RE((Sdr(e) \leftrightarrow u_r)\rvert$	31	# threads involving $(Sdr(e)$, u_r as senders
25	$\frac{\lvert RE((Sdr(e) \rightarrow u_r)\rvert}{\lvert E(u_r \rightarrow (Sdr(e))\rvert}$		
26	$\frac{\lvert RE(u_r \rightarrow (Sdr(e))\rvert}{\lvert E((Sdr(e) \rightarrow u_r)\rvert}$		

Table 5: Kendall τ distance (A^M,R^M).

M=	EC	ER	ET	TC	RG	RW
	0.46	0.52	0.49	0.46	0.5	0.11

correlation between l_1 and l_2 [5, 7].

Correlation between engagingness and Responsiveness. We first show the correlation between engagingness and responsiveness for each proposed model. Table 13 illustrates the Kendall τ distance of engagingness and responsiveness ordered lists from each model. The τ distance ranges between 0.4 and 0.5 for most models (except RW). These results indicate that engagingness and responsiveness are fairly distinctive behaviors. Most users would receive different ranks for engagingness and responsiveness.

Correlation between different models. Table 6 and Table 7 show the correlations of pairs of models by engagingness and responsiveness, respectively. Table 6 shows that the different engagingness models are quite similar, especially email count model (EC) and thread count model (TC)[3]. This is due to most email threads having two to three emails each. The similarity across different models is even more prominent for responsiveness as shown in Table 7. Again, the EC and TC models show high correlation in the responsiveness ranking. In particular, our proposed models are correlated by responsiveness rather than by engagingness. The email based models such as ER and ET are highly correlated in both engagingness and responsiveness. On the other hand, the random walk (WR) model appears to rank users more differently from all other models in both engagingness

[3] The most correlated entry is shown in boldface while entries < 0.05 are underlined.

Table 6: Kendall τ distance between engagingness models.

	A^{ER}	A^{ET}	A^{TC}	A^{RG}	A^{RW}
A^{EC}	0.14	0.16	**0.01**	0.18	0.22
A^{ER}		0.12	0.14	0.15	0.24
A^{ET}			0.16	0.15	0.22
A^{TC}				0.18	0.22
A^{RG}					0.24

Table 7: Kendall τ distance between responsiveness models.

	R^{ER}	R^{ET}	R^{TC}	R^{RG}	R^{RW}
R^{EC}	0.06	$\underline{0.03}$	**0.01**	$\underline{0.03}$	0.26
R^{ER}		0.07	0.06	0.08	0.25
R^{ET}			$\underline{0.03}$	$\underline{0.03}$	0.26
R^{TC}				$\underline{0.03}$	0.26
R^{RG}					0.27

and responsiveness. This is not a surprise due to its rather unique way of measuring behaviors.

Most engaging and responsive users. Table 8 shows the top five engaging users and top five responsive users after averaging the ranks of our proposed models. The table shows that the two sets of top users are different, consistent with our earlier results. It is interesting to note that most engaging users are traders. Other than CEO John Lavorato, the top responsive users are general employees.

Table 8: Top-5 users by engagingness and responsiveness.

	engagingness		Responsiveness	
Rank	Enron employee	Position	Enron employee	Position
1	Ryan Slinger	Trader	John Lavorato	CEO
2	Larry Campbell	N/A	Monika Causholli	Employee
3	Joe Quenet	Trader	Jeff Dasovich	Employee
4	Mike Swerzbin	Trader	Kate Symes	Employee
5	Jeff King	Manager	Kay Mann	Employee

Table 9: Results of email reply order prediction.

Features used in SVM	Average Accuracy (%)
$\mathrm{SVM}_{\mathbb{E}+\mathbb{I}}$	76.68
$\mathrm{SVM}_{\mathbb{U}}$	77.31
$\mathrm{SVM}_{\mathbb{B}}$	67.37
$\mathrm{SVM}'_{\mathbb{E}+\mathbb{I}}$	65.33
$\mathrm{SVM}'_{\mathbb{U}}$	69.78

2.4 Experiments - Email Reply Order Prediction Accuracy

Prediction performance. The goal of this experiment is to evaluate the performance our proposed classification approach to predict email reply order. We also want to examine the usefulness of engagingness and responsiveness behaviors in prediction task. There are five SVM classifiers trained, namely: (a) using comparative email and interactive features (denoted by $\mathrm{SVM}_{\mathbb{E}+\mathbb{I}}$); (b) using comparative behavior features only (denoted by $\mathrm{SVM}_{\mathbb{B}}$), (c) using all features (denoted by $\mathrm{SVM}_{\mathbb{U}}$), (d) using comparative email and interactive features except $t(e)$ (denoted by $\mathrm{SVM}'_{\mathbb{E}+\mathbb{I}}$), and (e) using all features except $t(e)$ (denoted by $\mathrm{SVM}'_{\mathbb{U}}$). Classifiers (d) and (e) are included as earlier study has shown that email replies often follow the last-in-first-out principle. $\mathrm{SVM}'_{\mathbb{E}+\mathbb{I}}$ and $\mathrm{SVM}'_{\mathbb{U}}$ allow us to find out if we can predict without knowing the email time information.

From the 27,730 email reply relationships, we extracted a total of 19,167 email pairs for the prediction task. The emails in each pair have replies that comes after the two emails are received by the same user. For each email pair, we computed feature values based on only email data occurred before the pair. In addition, we used complement email pairs in training. The complement of an email pair (e_i,e_j) with class label c is another email pair (e_j,e_i) with class label $\bar{c}$. Five folds cross validation was used to measure the average accuracy of the classifiers over the five folds. The accuracy measure is defined by $\frac{\#\ correctly\ classified\ pairs}{\#\ email\ pairs}$.

Figure 9 illustrates the results of all the five SVM classifiers. $\mathrm{SVM}_{\mathbb{U}}$ produces the highest accuracy of 77.31% due to the use of all available features. By excluding the email arrival order feature, the accuracy (of $\mathrm{SVM}'_{\mathbb{U}}$) reduces to 69.78%. This performance is reasonably good given that random prediction gives an accuracy of 50%. The classifier using behavior features only ($\mathrm{SVM}_{\mathbb{B}}$) is 2% more accurate than that with email and interaction features without email arrival order feature ($\mathrm{SVM}'_{\mathbb{E}+\mathbb{I}}$). The above results show that email arrival order feature is an important feature in the prediction

Table 10: Top-10 features for $\text{SVM}'_{\mathbb{U}}$.

Rank	Feature	Weight
1	$A^{ET}(Sdr(e_i)) - A^{ET}(Sdr(e_j))$	0.66
2	$R^{RG}(Sdr(e_i)) - R^{RG}(Sdr(e_j))$	0.57
3	$Indegree(Sdr(e_i)) - Indegree(Sdr(e_j))$	0.54
4	$A^{RW}(Sdr(e_i)) - A^{RW}(Sdr(e_j))$	0.53
5	# threads involving u_i, u_j *as senders*	0.47
6	$R^{TC}(Sdr(e_i)) - R^{TC}(Sdr(e_j))$	0.46
7	$A^{ER}(Sdr(e_i)) - A^{ER}(Sdr(e_j))$	0.39
8	$\lvert E(Sdr(e_i) \rightarrow u_r) - E(Sdr(e_j) \rightarrow u_r)$	0.28
9	$size(r(e_i)) - size(r(e_j))$	0.27
10	$A^{RG}(Sdr(e_i)) - A^{RG}(Sdr(e_j))$	0.24

task. We however notice that behavior features contribute to prediction accuracy especially when the email arrival order feature is not available.

Top features. Table 10 depicts the top 10 features for the $\text{SVM}_{\mathbb{U}}$ classifier. The table shows that engagingness based on the email reply time model RT is the most discriminative feature. Seven out of ten top features are behavior features. This suggests that engagingness and responsiveness are useful in predicting email reply order.

2.5 Discussions

In this paper, we formulate the user engagingness and responsiveness behaviors in an email network. We have developed six behavior models based on different principles. Using the Enron data set, we evaluate these models. We also apply the models to email reply order prediction task and demonstrate that behavior features can be useful in this task. The work is a significant step beyond the usual node and network statistics to determine user behaviors from their interactions. While our results are promising, there are still much room for further research. Firstly, behaviors are mutually dependent and we plan to introduce mutual dependency into our models. Secondly, behaviors can be localized as a user may not behave the same towards different users. Some users may be more responsive to friends than strangers. The localized behavior models should therefore be explored.

3 Behavior Modeling in Mobile Social Networks

Mobile social networks are gaining popularity with the pervasive use of mobile phones and other handheld devices. In these networks, users maintain friendship links, exchange short messages and share content with one another. From the social communication standpoint, messaging in mobile social networking supplements the existing face-to-face or phone communications as users establish social relationships with one another. Cummings, Butler and Kraut found that online social relationships are usually weaker than offline social relationships[3]. In their work, the online social relationships refer to those established through emailing. While we may generalize the results to relationships established through messaging, there is a lack of study to relate messaging behaviors with online relationships between users, and messaging behaviors with social status of users in an online community.

In this part of research, we study mobile messaging related user behaviors in myGamma, a well established mobile social networking site that supports both friendship links and messaging services. Again, we distinguish two types of user behaviors: soliciting active responses for an initiated message (or link) and responding to an incoming message (or link). The behaviors are also known as user *engagingness* and *responsiveness* respectively.

Our thesis in this work is that engagingness and responsiveness behaviors are related to the social status of users in a friendship network as well as their communication patterns with other users. We specifically aim to answer the following interesting research questions: (a) How can we tell if a user is engaging or responsive from his/her messaging activities? (b) How are a user's engagingness and responsiveness behaviors related to his/her status in friendship networks? (c) Are the messaging behaviors related to topics of messages? If so, what are the relationships like?

Modeling user behaviors can be challenging attributed to the wide variety of messages and the connectedness among users in the messaging networks. Messages can be categorized in numerous ways based on its formality, sentiments, and content. Instead of applying natural language text understanding techniques on the message content which is usually computationally costly and inaccurate, we want our messaging behavior models to be defined upon the messaging header data already available as well as the ways (friendship links) users are linked to one another. As one's behaviors can be affected by all his/her neighbors, the messaging behavior models should be able to cope with all the inter-dependency between behaviors.

Mobile messaging in many ways are similar to instant messaging popular among web users. Both support real-time synchronous communications whenever users are online. Mobile messaging however has the additional feature of storing incoming messages whenever users are offline so that the messages can be read when the users become online again. Such a feature enables mobile messaging to behave like email messaging which supports mainly asynchronous communications. As noted in [9], instant messaging users are likely to communicate with few acquainted users as opposed to strangers. Mobile messaging is also different from instant messaging by not restricting the communicating users to be friends on a user's contact list.

The above differences have therefore distinguished our work from the previous works that focus on instant messaging. To the best of our knowledge, engagingness and responsiveness are behaviors yet to be studied in mobile social networks, particularly in large scale. The work presented in this paper is thus early efforts in this direction. Messaging behaviors of users during online and offline periods can be different yet related. In this paper, we demonstrate that a user's online (and offline) durations can be estimated from the time of messages sent by him/her. From the online durations, we derive the online and offline messaging sessions between users which are in turn used to define the online and offline messaging behaviors.

Our contributions can be summarized as follows:

- We propose several quantitative models for measuring user engagingness and responsiveness in both online and offline messaging sessions. These include the MsgCount, ReplyTime, SessionInit and Sequence models. We further extend these models to incorporate mutual dependency between engagingness and responsiveness.

- We apply these models on a myGamma dataset containing both messages and friendship links between users. Comparisons between engagingness and responsiveness, and comparisons between different models have been made using this real dataset. We further relate the two behaviors with number of friendships users enjoy.

- We finally show that engaging and responsive users play important roles in messaging topics within an online community. We apply Latent Dirichlet Allocation [2] to uncover latent topics from our message dataset. We discover that major topics in the community are driven by engaging and responsive users.

3.1 Related Work

Synchronous vs asynchronous messaging. Messaging is a mode of communication. Depending on whether the users in communication are physically present together and whether they are able to receive and respond messages in realtime, we can classify messaging services to be synchronous, asynchronous, or semi-synchronous. Instant messaging and email messaging are representatives of synchronous and asynchronous messaging respectively. Mobile messaging is more a mixture of both and is thus semi-synchronous. There are very few previous efforts on studying user behaviors in email messaging. In [4], user responsiveness behavior is defined in the context of replying emails of the same subject headings. In instant and mobile messaging, message structures are much simpler and subject heading is not longer a viable grouping criteria. This work does not cover the engagingness behavior nor explore different responsiveness behavior models. To the best of our knowledge, there is no other research on modeling messaging behaviors.

Instant messaging behaviors. As instant messaging is very similar to the myGamma's messaging, we examine related work in the area. Nardi, Whittaker and Bradner found that instant messaging serves largely social purposes instead of formal information exchanges even in the organization setting. Avrahami and Hudson studied the responsiveness of users in instant messaging[1]. The responsiveness here refers to the response time required for a user to respond to an incoming *session initiation attempt* (SIA) message. The SIA message is an incoming message from a sender that reaches a user long after, determined by some threshold, the user have sent the previous message to the sender. Strictly speaking, the responsiveness concept here is not a user behavior but some response time label. One of five response time labels are assigned to each message replied in 30 seconds, 1, 2, 5 and 10 minutes respectively, and the prediction models proposed could achieve 80 to 90% accuracy in assigning response time labels.

Unlike [1], we focus mainly on mobile messaging related user behaviors. Due to the peculiar nature of mobile messaging, we have to perform classification of online and offline periods for each user. Instead of treating responsiveness as message response time, we study responsiveness as a quantitative user characteristics. We also introduce engagingness as another user characteristics. Our work have also involved a much larger dataset.

3.2 Preliminaries

Mobile messaging users communicates with one another using a mixture of online and offline messaging sessions. When a user and his contact are online, they can exchange exchanges with each other in realtime. This mode of messaging is similar to instant messaging which supports highly synchronous communication. On the other hand, a mobile messaging user can also send messages to another user if the latter is offline. Such messages are stored and are retrieved when the recipient becomes online again. Such a messaging mode is more similar to emailing and text messaging which are representatives of asynchronous communication.

With both synchronous and asynchronous communication taking place in mobile messaging, a mixture of messaging behaviors can exists for the same users. To study these messaging behaviors separately, we take the following steps:

- Step 1 (determine the online and offline durations of users): Each user may be online or offline when using mobile messaging. For cases where online and offline durations of users are not logged, we need determine these durations automatically based on time gaps between consecutive messages. As every user has his/her messaging pattern, a personalized approach to determination of online and offline duration will be required. A detailed description of our proposed approach is given in Section 3.3.
- Step 2 (identify the online and offline messaging session between users): Once the users' online durations are determined, we proceed to derive the online and offline messaging sessions between every communicating pair of users (see Section 3.4). At the end of this step, each user pair may have zero or more online/offline sessions.

Table 11 defines the notations to be used in the rest of paper. A message m' is said to be the *reply* of a m if it is the earliest message that has $Sdr(m') = Rcp(m)$, $Rcp(m') = Sdr(m)$, and $t(m') > t(m)$.

3.3 Determination of Online and Offline Status

Determining the online and offline communication for mobile messaging users is a non-trivial task. In the absence of a log of user online status over time, we have resort to a statistical approach to automatically decide the online and offline periods of each user as he or she uses the messaging service.

Our main proposed idea of segmenting messages into online and offline messages is based on a **Gaussian Mixture Model**. In this model, we envisage that users send out messages at different rates depending on whether they are online or offline. We first define a random variable X for the time gap between two consecutive messages sent by all users. Assume that X is formed by two clusters of time gaps, i.e., online and offline. X can be modeled by a mixture of two Gaussian distributions $\mathcal{N}(\mu_1, \sigma_1^2)$ and $\mathcal{N}(\mu_2, \sigma_2^2)$

Table 11: Notations.

$SE(u_i)$	Messages sent by user u_i
$RE(u_i)$	Messages received by u_i
$RB(u_i)$	Messages replies sent by u_i
$RT(u_i)$	Messages replying to u_i's earlier messages
OnP_i	Online periods of u_i
OffP_i	Offline periods of u_i
$\mathbf{S}_{ij}$	Online sessions between u_i and u_j
$\bar{\mathbf{S}}_{ij}$	Offline sessions between u_i and u_j
$r(m)$	Reply to message m
$Sdr(m)$	Sender of message e
$Rcp(m)$	Recipient of message m
$t(m)$	Sent time of message m
$\mathbf{M}_{i \to j}$	Messages from u_i to u_j
$\mathbf{M}_{ij}$	Messages between u_i and u_j

where μ_1 and μ_2 represent the mean time gaps of the two distributions respectively, while σ_1 and σ_2 represent the standard deviations respectively. We want to learn these parameters that generate distributions fitting our dataset.

Suppose we have N number of observed samples. Let x_n denote the n^{th} observed sample and $\mathcal{N}(x_n; \mu_k, \sigma_k^2)$ denote the probability that x_n is in cluster k. Let $\pi_k \in [0, 1]$ be the size of cluster k. We use EM algorithm to solve for the values of π_k, μ_k and σ_k as follows:

$$f(n,k) = \frac{\pi_k \mathcal{N}(x_n; \mu_k, \sigma_k^2)}{\sum_{j=1}^{2} \pi_j \mathcal{N}(x_n; \mu_j, \sigma_j^2)} \tag{14}$$

$$\pi_k = \frac{\sum_{n=1}^{N} f(n,k)}{N} \tag{15}$$

$$\mu_k = \frac{\sum_{n=1}^{N} x_n f(n,k)}{\sum_{n=1}^{N} f(n,k)} \tag{16}$$

$$\sigma_k^2 = \frac{\sum_{n=1}^{N} f(n,k)(x_n - \mu_k)^2}{\sum_{n=1}^{N} f(n,k)} \tag{17}$$

Once the parameters are learnt, the Gaussian distribution with smaller μ_k models the time gaps between send messages when users are in online periods while another Gaussian distribution models the time gaps when users are in offline periods. We also derive a *time gap threshold* γ to easily classify time gaps into online and offline periods.

3.4 Online and Offline Sessions

A message session s between two users u_i and u_j is defined by a set of consecutive messages between them. Due to the different online and offline messaging behaviors, we further divide sessions into online and offline sessions.

Given a set of messages $\mathbf{M}_{ij}$ between u_i and u_j, and the online periods of u_i and u_j denoted by $OnP_i = \{[ts_{i1}, te_{i1}], \cdots, [ts_{ik_i}, te_{ik_i}]\}$ and $OnP_j = \{[ts_{j1}, te_{j1}], \cdots, [ts_{jk_j}, te_{jk_j}]\}$ respectively.

The set of overlapping online periods between u_i and u_j, P_{ij}, is defined by:

$$\begin{aligned} OlpP_{ij} &= OnP_i \cap OnP_j \\ &= \{[max(ts_i, ts_j), min(te_i, te_j)] | [ts_i, te_i] \in OnP_i, \\ & \quad [ts_j, te_j] \in OnP_j, (ts_i > te_j) \wedge (ts_j > te_i)\} \end{aligned}$$

The set of online sessions between u_i and u_j, $\mathbf{S}_{ij}$, is then defined as a collection of message sets induced by the overlapping online periods such that each message set consists of at least some exchange of messages between u_i and u_j.

$$\begin{aligned} \mathbf{S}_{ij} &= \{\mathbf{M}_{ij}(p) | p \in OlpP_{ij} \wedge \\ & \quad (\exists m, m' \in \mathbf{M}_{ij}(p), m' = r(m))\} \end{aligned}$$

where $\mathbf{M}_{ij}(p) = \{m \in \mathbf{M}_{ij} | t(m) \in p\}$.

The set of online session intervals between u_i and u_j, $OnSsnP_{ij}$, is thus the set of overlapping online periods that cover online sessions, i.e.:

$$OnSsnP_{ij} = \{p \in OlpP_{ij} | \exists m, m' \in \mathbf{M}_{ij}(p), m' = r(m)\}$$

From the online session intervals, we derive the remaining periods as:

$$RemP_{ij} = [min(ts_i^*, ts_j^*), max(te_i^*, ts_j^*)] - OnSsnP_{ij}$$

where ts_i^* (ts_j^*) and te_i^* (te_j^*) denote the minimum ts_i (ts_j) and maximum te_i (te_j), respectively, in OnP_i (OnP_j).

The set of offline sessions $\bar{\mathbf{S}}_{ij}$ is then defined as a collection of message sets induced by the remaining periods such that each message set consists of at least some exchange of messages between u_i and u_j.

$$\begin{aligned} \bar{\mathbf{S}}_{ij} &= \{\mathbf{M}_{ij}(p) | p \in RemP_{ij} \wedge \\ & \quad (\exists m, m' \in \mathbf{M}_{ij}(p), m' = r(m))\} \end{aligned}$$

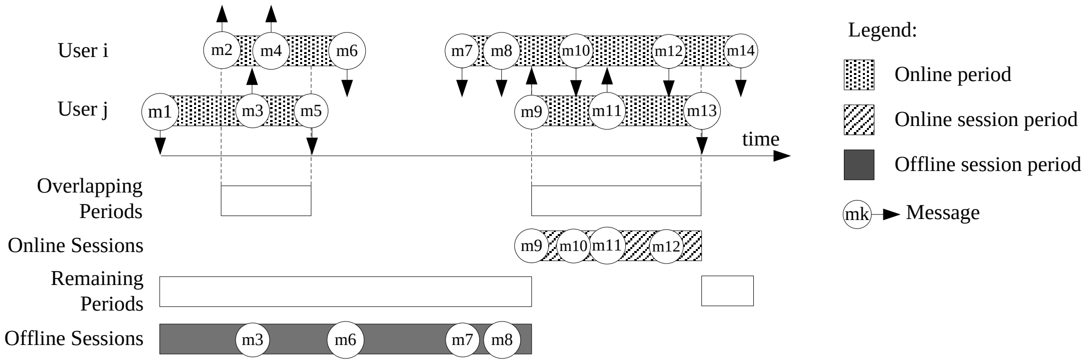


Figure 5: Online/Offline Periods and Sessions

The set of online session intervals between u_i and u_j, $\mathit{OffSsnP}_{ij}$, is thus the set of remaining periods that cover online sessions, i.e.:

$$\mathit{OffSsnP}_{ij} = \{p \in \mathit{RemP}_{ij} | \exists m, m' \in \mathbf{M}_{ij}(p), m' = r(m)\}$$

The start and end times of a session s refer to the times of the first and last messages respectively. The user who sends the first message of s is also known as the *initiator* of the session.

Consider the example shown in Figure 5. Users u_i and u_j have two online periods each. The messages directed between them are the ones exchanged between u_i and u_j. The messages directed away from them are sent to other users. Although u_i and u_j are both online in the left overlapping period, it does not constitute an online session due to a lack of message exchange between them. The only online session between u_i and u_j is thus $\{m_9, m_{10}, m_{11}, m_{12}\}$. Among the two remaining periods, only the left one has message exchanges between u_i and u_j. Hence, the offline session found is $\{m_3, m_6, m_7, m_8\}$.

3.5 Mobile Social Network Dataset

In the myGamma mobile social networking site, members interact and form online communities. Most members are young adults between the age of 20 to 30. The myGamma dataset we obtained consists of 194,809 users and 2.7M messages among them within the one-month period from September 8, 2009 to September 10, 2009. We first selected the users with at least one friendship link as not all users specify their friendships. Other than

Table 12: Dataset Statistics.

Users	14,423
Messages	1,441,272
Sessions	72,297
Online sessions	5,491
Offline sessions	66,806
Users participating sessions	10,346
Users participating online sessions	4,441
Users participating offline sessions	10,096
Users initiating sessions	9,408
Users initiating online sessions	3,035
Users initiating offline sessions	9,186
Messages in sessions	199,073
Messages in online sessions	12,318
Messages in offline sessions	186,755
Friendship links	1,795,674
Foe links	109,510
Message links	1,196,011

friendship network, we have message links between users forming the message network. A message link from user u_i to user u_j is defined when there is at least one message from u_i to u_j. We further selected the users who have sent at least 4 messages and received at least 4 messages. This way, we obtained a final dataset with 14,423 users with 1,196,011 messages among them. Within this set of messages, 236,798 are replies to some messages in the set. Table 12 summarizes the statistics of this final dataset.

We apply Gaussian Mixture Model on the dataset to determine the online and offline periods of users. To avoid bias against time gap threshold introduced by users who send very few (one or two) messages, we sample the time gaps from users who have at least 100 messages each. There are 3520 such users. The time gap threshold γ obtained is around 4 hours (see Figure 6). The threshold is subsequently applied to the final dataset to obtain online and offline sessions with numbers shown in Table 12.

3.6 User engagingness and Responsiveness for Mobile Messaging

In this section, we will introduce four pairs of basic engagingness and responsiveness behavior models, namely MsgCount, ReplyTime, SessionInit, and Sequence. They are designed based on message, reply time, session

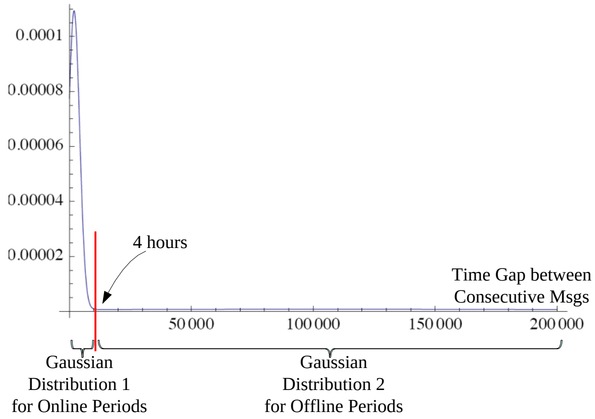


Figure 6: Two Gaussian Mixture Model for Determining Online and Offline Periods in MyGamma Dataset

and messaging sequence data respectively. Each model assigns an engagingness (responsiveness) score $\in [0, 1]$ to each user, 0 for non-engaging (non-responsive) user and 1 for fully engaging (fully responsive) user. As users may demonstrate different messaging behaviors during online and offline sessions, every model *except* SessionInit has both online and offline versions. For example, the online and offline session versions of MsgCount are MsgCount$_{on}$ and MsgCount$_{off}$ respectively. For SessionInit model, only the online version is applicable as it involves the online sessions only.

MsgCount Model: This model is designed based on the principle that an engaging user should have most of his/her messages replied by other users, while a responsive user should have most of his/her received messages replied. The engagingness and responsiveness scores, A^{MsgCount} and R^{MsgCount}, for online and offline sessions are thus defined by:

$$A_x^{\text{MsgCount}}(u_i) = \frac{|RT_x(u_i)|}{|SE_x(u_i)|} \tag{18}$$

$$R_x^{\text{MsgCount}}(u_i) = \frac{|RB_x(u_i)|}{|RE_x(u_i)|} \tag{19}$$

where session type x can be online or offline denoted by on and off respectively.

ReplyTime Model: Unlike MsgCount, this model examines the reply times of messages to determine user engagingness and responsiveness. An engaging user should have his/her messages quickly replied by others while a responsive user should have received messages quickly replied. Given a message m' which is a reply of message m, i.e., $m' = r(m)$, the *reply time* of m', is $rt(m') = t(m') - t(m)$. The z-normalized reply time $\hat{rt}(m')$ is defined by $\frac{rt(m')-\overline{rt}}{\sigma_{rt}}$ where $\overline{rt}$ and σ_{rt} are the mean and standard deviation of reply time respectively. Now, we define the engagingness and responsiveness of ReplyTime model as:

$$A_x^{\textsc{ReplyTime}}(u_i) = \frac{1}{|SE_x(u_i)|} \sum_{\substack{m \in SE_x(u_i) \\ m'=r(m)}} f(\hat{rt}(m')) \tag{20}$$

$$R_x^{\textsc{ReplyTime}}(u_i) = \frac{1}{|RE_x(u_i)|} \sum_{\substack{m \in RE_x(u_i) \\ r(m)=m'}} f(\hat{rt}(m')) \tag{21}$$

where

$$f(x) = \frac{e^{-x}}{1+e^{-x}} \tag{22}$$

The function $f()$ is designed to convert the normalized reply time to the range [0,1] with 0 and 1 representing extreme slow and extreme fast reply times respectively.

SessionInit Model: In this model, we adopt the principle that an engaging user is more likely to initiate online messaging sessions for the messages he/she sends out, while a responsive user is more likely to participate in online sessions initiated by messages from others. We first denote the number of online session initiating and participating messages of a user u_i by $SsnInitMsg(u_i)$ and $SsnMsg(u_i)$ respectively. These are the first messages of online sessions. SessionInit Models for engagingness and responsiveness are then defined as:

$$A_{on}^{\textsc{SessionInit}}(u_i) = \frac{|SsnInitMsg(u_i)|}{|SsnInitMsg(u_i)| + |SE_x(u_i) - SsnMsg(u_i)|} \tag{23}$$

$$R_{on}^{\textsc{SessionInit}}(u_i) = \frac{\sum_j |SsnInitMsg(u_j) \cap \mathbf{M}_{j\to i}|}{\sum_j |SsnInitMsg(u_j) \cap \mathbf{M}_{j\to i}| + |\mathbf{M}_{j\to i} - SsnMsg(u_j)|} \tag{24}$$

where $SsnInitMsg(u_j) \cap \mathbf{M}_{j\to i}$ represents the set of messages from u_j to u_i that successfully initiate online sessions with u_i, and $\mathbf{M}_{j\to i} - SsnMsg(u_j)$

represents the set of messages from u_j to u_i that fails to initiate online sessions with u_i.

Sequence Model. Message sequence refers to the sequence of messages sent and received by a user ordered by time. To derive engagingness and responsiveness from message sequences, we consider the principle that an engaging user is expected to have his or her sent messages replied soon after they are received by the message recipient, and a responsive user replies soon after they receive messages. As the time taken to reply an message may vary, we consider the number of messages received later than a message m but are replied before m by a user as a proxy of how soon m is replied.

The above principle is thus used to develop the Sequence Model. Let $seq_{x,i}$ denote the online ($x = on$) or offline ($x = off$) session message sequence of user u_i. When a message received by u_i is replied before other message(s) received earlier, the reply of the former is known as an *out-of-order reply.* Formally, for a message m received by u_i, we define the *number of messages received* and *number of out-of-order replies* between m and its reply m' in $seq_{x,i}$, denoted by $n_{x,r}(u_i, m)$ and $n_{x,\overline{o}}(u_i, m)$ respectively, as

$$n_{x,r}(u_i,m) = \begin{cases} \text{\# messages received between} & \text{if } \exists m' \in RT_x(u_i), \\ m \text{ and } m' \text{ in } seq_{x,i}, & \quad r(m) = m' \\ -1, & \text{otherwise} \end{cases} \tag{25}$$

$$n_{\overline{o}}(u_i,m) = \begin{cases} \text{\# messages received} & \text{if } \exists m' \in RT_x(u_i), \\ \text{ between } m \text{ and } m' \text{ in } seq_{x,i} & \quad r(m) = m' \\ \text{ and have been replied,} & \\ -1, & \text{otherwise} \end{cases} \tag{26}$$

The -1 value is assigned to $n_{x,r}$ and $n_{x,\overline{o}}$ when m is not replied at all. The user engagingness and responsiveness of the Sequence_x model are thus defined as:

$$A_x^{\textsc{Sequence}}(u_i) = \frac{\sum_{m \in SE_x(u_i), u_j = Rcp(m)} (1 - \frac{n_{x,\overline{o}}(u_j,m)}{n_{x,r}(u_j,m)})}{|SE_x(u_i)|} \tag{27}$$

$$R_x^{\textsc{Sequence}}(u_i) = \frac{\sum_{m \in RE_x(u_i)} (1 - \frac{n_{x,\overline{o}}(u_i,m)}{n_{x,r}(u_i,m)})}{|RE_x(u_i)|} \tag{28}$$

3.7 Mutual Dependency Based Models

In the above basic models, user engagingness and responsiveness are computed independently. They share the same underlying assumption that messaging behaviors of a user is independent of other users. This assumption

Table 13: Correlation of engagingness models in online sessions.

	A^{RT}	A^{SI}	A^{SQ}	$A^{\text{MC}*}$	$A^{\text{RT}*}$	$A^{\text{SI}*}$	$A^{\text{SQ}*}$
A^{MC}	0.86	0.98	0.99	0.86	0.86	0.73	0.86
A^{RT}		0.85	0.86	0.99	0.99	0.79	0.99
A^{SI}			0.98	0.85	0.85	0.75	0.85
A^{SQ}				0.86	0.86	0.74	0.86
$A^{\text{MC}*}$					0.99	0.79	0.99
$A^{\text{RT}*}$						0.79	0.99
$A^{\text{SI}*}$							0.79

Table 14: Correlation of responsiveness models in online sessions.

	R^{RT}	R^{SI}	R^{SQ}	$R^{\text{MC}*}$	$R^{\text{RT}*}$	$R^{\text{SI}*}$	$R^{\text{SQ}*}$
R^{MC}	0.85	0.98	0.99	0.86	0.85	0.97	0.86
R^{RT}		0.81	0.86	0.99	0.99	0.88	0.99
R^{SI}			0.98	0.81	0.81	0.99	0.81
R^{SQ}				0.86	0.86	0.97	0.86
$R^{\text{MC}*}$					0.99	0.88	0.99
$R^{\text{RT}*}$						0.88	0.99
$R^{\text{SI}*}$							0.88

does not always hold in practice as user behaviors are likely to be affected by other users he or she communicates with. Hence, we have designed the mutual dependency based engagingness and responsiveness models.

Suppose $A^M(u_i)$ and $R^M(u_i)$ are engagingness and responsiveness of user u_i computed using model M. The mutual dependency between A^M and R^M can be expressed as:

- A user is considered more engaging if he/she can get less responsive users to respond. Formally, we write:

$$A^{M*}(u_i) = \frac{\sum_{u_j} v^M_{u_i,u_j} \cdot (1 - R^M(u_j))}{|SE_x(u_i)|} \tag{29}$$

- A user is considered more responsive if he/she responds to less engaging users.

$$R^{M*}(u_i) = \frac{\sum_{u_j} w^M_{u_i,u_j} \cdot (1 - A^M(u_j))}{|RE_x(u_i)|} \tag{30}$$

where $v^M_{u_i,u_j}$ and $w^M_{u_i,u_j}$ denote the quantity values between u_i and u_j computed based on the principle of M (i.e., # of replies between u_i and u_j in $A^{\text{MC}}_x(u_i)$).

Table 15: Correlation of engagingness and responsiveness models in online sessions.

Model	Spearman's rho	Model	Spearman's rho
MC	0.83	$MC*$	0.75
RT	0.75	$RT*$	0.75
SI	0.78	$SI*$	0.72
SQ	0.83	$SQ*$	0.75

4 Experiment Results - Comparison of Messaging Behaviors

For comparison between user behavior models, we compare by examining Spearman's rank correlation coefficient. The Spearman's rho of two ranked list l_1 and l_2, $\rho(l_1, l_2)$ is defined by:

$$\rho(l_1, l_2) = 1 - \frac{6 \sum d_{u_i}^2}{n(n^2 - 1)} \tag{31}$$

where l_1 and l_2 have n users' ranks and the difference $d_{u_i} = l_1(u_i) - l_2(u_i)$ between the ranks of user u_i on l_1 and l_2. ρ value falls between -1 and 1 representing negative correlation and positive correlation respectively. In addition, $\rho = 0$ stands for no linear correlation.

Comparison between user engagingness (responsiveness) models. Table 13 (Table 14) shows the *Spearman's rho* between the ranked lists produced by different engagingness (responsiveness) models for online sessions. The table shows that most engagingness (responsiveness) models are very similar to one another except A^{SI} and A^{SI^*} which are slightly more different. This is because of the principle of the SessionInit Model which is distinct from the other models. In the SessionInit Model, the engagingness of a user will be high when the user tends to initiate a number of sessions. However, it turns out that most users usually initiate a small number of sessions in the myGamma dataset. Though not shown here, we also observe the same for engagingness (responsiveness) in offline sessions.

Comparison between engagingness and responsiveness. Next, we examine the difference between engagingness and responsiveness for different models for online sessions. As shown in Table 15, the Spearman's rho values between the two behaviors of the same model are mostly more different than differences observed between two models for the same behavior (say, engagingness). The only exception is SessionInit model. This can be relatively sparser data for measuring the model. Interestingly, for offline sessions, we observe that the distinction between engagingness and

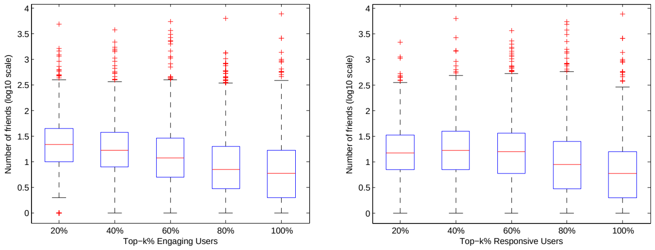


Figure 7: Engagingness/responsiveness and friendship links.

responsiveness is less obvious. This could be due to offline nature (i.e., long time lag) of responding messages between users.

Engagingness/responsiveness and friendship links Figure 7 depicts the boxplots of number of bi-directed friendship links of users divided into five different engagingness/responsiveness intervals of size 0.2. Here, we derive the overall engagingness (responsiveness) of each user by averaging the engagingness (responsiveness) of different models (including online and offline versions). We observe that users with higher engagingness have more friendship links. This is less obvious for responsiveness. This suggests that engaging users are more capable of attracting and establishing friendships.

5 Experiment Results - Topic Specific Messaging Behavior Analysis

5.1 Motivation

Users demonstrate different messaging behaviors in different topics of discussion. For interesting topics, one expect users to be more engaging and responsive, while uninteresting topics will only turn users away from participation. In this section, we analyze user engagingness and responsiveness for different message topics in our dataset. The purpose here is to identify interesting topics within the online community.

To conduct this study, we first identify the major message topics from the aggregated message content for a set of users using Latent Dirichlet Allocation (LDA) [2]. We then analyze the distribution of engagingness and responsiveness of users within each message topic.

Table 16: Major Topics.

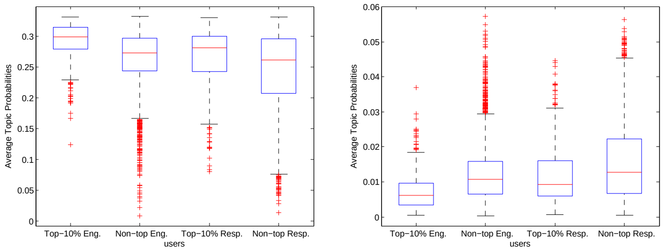
Topics	Top 10 terms
T14	love, chat, hello, want, dear, baby, friend, dont, hope, miss
T15	dear, chat, sana, sawa, doin, kwani, swty, pliz, thea, sasa,
T17	view, blkapp, mode, click, gift, return, gifts, love, private, thank

5.2 Message Topic Distillation

For our analysis purpose, we only select users indicating English as their preferred language and there are only 27,920 such users. Despite this pruning effort, there are still some users writing non-English messages as shown in our results. Due to the limited content in each message, we aggregate the messages by their senders and recipients. Messages sent by a user capture the topics in which he/she is interested to communicate with others. On the other hand, messages received by a user represent the topics about which others wish to communicate with him/her. We call the two aggregated message content the out-document and in-document of the user. We also remove stop words from these content using a combined dictionary of 400+ stop words from [6]. Given a set of documents and k topics, LDA essentially finds the k latent topics in the documents such that each document is assigned a topic distribution, and each word occurrence in the document is assigned a topic. Since topics are not given beforehand, we performed LDA on the merged set of out-documents and in-documents with $k = 20$ common topics. The empirical choice of $k = 20$ appears to work well as we could find the popular topics exist in the data.

The topic distillation results are shown in Table 16. A uniform topic distribution assumption for users would have 0.1 assigned for each topic. Among the 20 topics, most have only a few hundreds of users (e.g., topic 1 has 141 users), while topics 14, 15, and 17 have 27,741, 17,088, and 4,780 users respectively. We call these users the main users. We empirically select topics 14, 15 and 17 as the major topics as they have much more main users. The remaining topics are thus the non-major topics.

To conserve space, we only show the top 10 terms found in the three major topics. Topic 14, the largest topic in term of main user count, consists of mainly greeting terms. This is not a surprise as users tend to greet one another in such a social network. Topic 15 appears to be dominated by abbreviated (e.g., “doin”=“doing”, “swty”=“sweety”) and non-English terms (e.g., “sana”, “sewa”, “kwani”). Topic 17 is likely to be related to use of software and exchange of gifts.



(a) Major Topics (b) Non-Major Topics

Figure 8: Average Topic Probability Distribution.

5.3 Messaging Behaviors in Message Topics

We would now like to examine the distinction between engaging (or responsive) users and other users in both major and non-major topics.

Figure 8a shows the boxplots of top 10% engaging (responsive) users' average major topic probabilities and those of non-top engaging (responsive) users. The average major topic probability of a user is derived by averaging the topic probabilities of his/her out-documents (in-documents) for the major topics (i.e., Topics 14, 15 and 17). Similarly, we derive the average non-major topic probability of each user in Figure 8b. Figure 8a shows that the top 10% engaging users contribute more to the major topics than the other users. On the other hand, the former contribute less on average to the non-major topics than the other users as shown in Figure 8b. From the figures, we also observe the major topics enjoy more user contribution than non-major topics in general. We also examine the average topic probability of top 10% responsive users and non-top 10% responsive users for major topics and non-major topics in Figure 8 showing similar results to engaging users. On the whole, the results match our intuition that engaging and responsive users are the ones driving important topics in the online community. That is, the former tends to generate messages of major topics while the latter tends to receive messages of major topics.

6 Conclusions

The project has so far examined engagingness and responsiveness behaviors in two datasets. It also resulted in two publications [11, 10].

References

[1] Daniel Avrahami and Scott E. Hudson. Responsiveness in instant messaging: predictive models supporting inter-personal communication. In *SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 731–740, 2006.

[2] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[3] Jonathon N. Cummings, Brian Butler, and Robert Kraut. The quality of online social relationships. *Communications of the ACM*, 45(7):103–108, 2002.

[4] P. Deepak, D. Garg, and V. Varshney. Analysis of Enron Email Threads and Quantification of Employee Responsiveness. In *Workshop on Text Mining and Link Analysis (TextLink 2007)*, 2007.

[5] R. Fagin, R. Kumar, and D. Sivakumar. Comparing Top K Lists. *SIAM J. on Discrete Mathematics*, 17:134–160, 2003.

[6] S. Howard, H. Tang, M. Berry, and D. Martin. GTP: General Text Parser. In *http://www.cs.utk.edu/∼lsi/*, 2009.

[7] M. Kendall. *Rank Correlation Methods.* Charles Griffin and Company Limited, 1948.

[8] N. E. Miller and J. Dollard. *Social Learning and Imitation.* Yale University Press, 1941.

[9] Bonnie A. Nardi, Steve Whittaker, and Erin Bradner. Interaction and outeraction: instant messaging in action. In *ACM conference on Computer supported cooperative work (CSCW)*, pages 79–88, 2000.

[10] Byung-Won On, Ee-Peng Lim, Jing Jiang, Freddy Chong Tat Chua, Viet-An Nguyen, and Loo-Nin Teow. Messaging behavior modeling in mobile social networks. In *Symposium on Social Intelligence and Networking (SIN-10)*, Minneapolis, USA, August 2010.

[11] Byung-Won On, Ee-Peng Lim, Jing Jiang, Amruta Purandare, and Loo-Nin Teow. Mining interaction behaviors for email reply order prediction. In *International Conference on Social Networks Analysis and Mining (ASONAM)*, Odense, Denmark, August 2010.