A Network Model for Dynamic Textual Communications with Application to Government Email Corpora

Bomin Kim
Department of Statistics
Pennsylvania State University

Aaron Schein
College of Information and Computer Sciences
University of Massachusetts Amherst

Bruce Desmarais
Department of Political Science
Pennsylvania State University

Hanna Wallach
Microsoft Research NYC
University of Massachusetts Amherst

Abstract

We introduce the interaction-partitioned topic model (IPTM)—a probabilistic model for who communicates with whom about what, and when. Broadly speaking, the IPTM partitions time-stamped textual communications, according to both the network dynamics that they reflect and their content. To define the IPTM, we integrate a dynamic version of the exponential random graph model and latent Dirichlet allocation The IPTM assigns each topic to an "interaction pattern"—a generative process for ties that is governed by a set of dynamic network features. Each communication is then modeled as a mixture of topics and their corresponding interaction patterns. We use the IPTM to analyze emails sent between department managers in Dare county government in North Carolina, and demonstrate that the model is effective at predicting and explaining continuous-time textual communications.

1 Introduction

In recent decades, real-time digitized textual communication has developed into a ubiquitous form of social and professional interaction (see, e.g., ????). From the perspective of the computational social scientist, this has lead to a growing need for methods of modeling interactions that manifest as text exchanged in continuous time. A number of models that build upon topic modeling through Latent Dirichlet Allocation (?) to incorporate link data as well as textual content have been developed recently (???). These models are innovative in their extensions that incorporate network tie information. However, none of the models that are currently available in the literature integrate the rich random-graph structure offered by state of the art models for network structure—in particular, the exponential random graph model (ERGM) (???). The ERGM is the canonical model for network structure, as it is flexible enough to specify a generative model that accounts for nearly any pattern of tie formation (e.g., reciprocity, clustering, popularity effects) (?). We build upon recent extensions of ERGM that model time-stamped ties (??), and develop the interaction-partitioned topic model (IPTM) to simultaneously model the network structural patterns that govern tie formation, and the content in the communications.

ERGM, and models based on ERGM, provide a framework for explaining or predicting ties between nodes using the network sub-structures in which the two nodes are embedded (e.g., an ERGM specification may predict ties between two nodes that have many shared partners). ERGM-style models have been used for many applications in which the ties between nodes are annotated with text. The text, despite providing rich information regarding the strength, scope, and character of the ties, has been largely excluded from these analyses, due to the inability of ERGM-style models to incorporate textual attributes of ties. These application domains include, among other applicaitons, the study of legislative networks in which networks reflect legislators' co-support of bills, but exclude bill text (??); the study of alliance networks in which net-

Preliminary work. Under review by AISTATS 2018. Do not distribute.

works reflect countries' co-signing of treaties, but exclude treaty text (**????**); the study of scientific coauthorship networks that exclude the text of the coauthored papers (**???**); and the study of text-based interaction on social media (e.g., users tied via 'mentions' on twitter) (**???**).

In defining and testing the IPTM we embed two core conceptual properties, in addition to modeling both text and network structure. First, we link the content component of the model, and network component of the model such that knowing who is communicating with whom at what time (i.e., the network component) provides information about the content of communication, and vice versa. Second, we provide fixible parameterization in modeling the timing of documents using generalized linear model approach. In what follows we (1) present the generative process for the IPTM, describing how it meets our theoretical criteria, (2) derive the sampling equations for Bayesian inference, and (3) illustrate the IPTM through application to email corpora of internal communications by government officials in Dare County, NC.

2 Model Definition

To define the IPTM, we begin by describing a probabilistic process by which documents are generated, where documents include author, recipients, contents, and timing. We provide a fully parametric definition of each component of the generative process, which enables the model to be used to simulate distributions of who communicates with whom about what, and when.

The data generated under the IPTM consists of D unique documents. A single document, indexed by $d \in \{1, \dots, D\}$, is represented by the four components $(a_d, \boldsymbol{r}_d, t_d, \boldsymbol{w}_d)$. The first two are the author and recipients of the document: an integer $a_d \in \{1, \dots, A\}$ indicates the identity of the author and a binary vector $\boldsymbol{r}_d = \{u_{di}\}_{i=1}^{A}$, which indicates the identity of the receipients. Next, t_d is the timestamp of the document d. For simplicity, we assume that documents are ordered by time such that $t_d < t_{d+1}$ for $d = 1, \dots, D$. Lastly, $\boldsymbol{w}_d = \{w_{dn}\}_{n=1}^{N_d}$ is a set of tokens that comprise the text of the document, where N_d denotes the total number of tokens in a document.

2.1 Content Generating Process

In this section, we illustrate how the words $\boldsymbol{w}_d$ are generated according to latent Dirichlet allocation (**?**). First, we generate the corpus-wide global variables that describe the content via topics.

For each topic $k = 1, \dots, K$:

- Choose a discrete distribution over V word types

$$\boldsymbol{\phi}_k \sim \text{Dirichlet}\Big(\beta, (\frac{1}{V}, \dots, \frac{1}{V})\Big).$$

Next, given that the number of words N_d is known, we generate each token by drawing a topic from the document-topic distribution and then drawing a word from the chosen topic.

For each document $d = 1, \dots, D$:

- Choose a discrete distribution over K topics

$$\boldsymbol{\theta}_d \sim \text{Dirichlet}\Big(\alpha, (m_1, \dots, m_K)\Big).$$

- For $n = 1, \dots, N_d$:
  - Choose a topic $z_{dn} \sim \text{Multinomial}(\boldsymbol{\theta}_d)$.
  - Choose a word $w_{dn} \sim \text{Multinomial}(\phi_{z_{dn}})$.

2.2 Interaction Patterns

They key idea that combines the IPTM component modeling "what" with the component modeling "who," "whom," and "when" is that different topics are associated with different interaction patterns. Each interaction pattern is characterized by a set of dynamic network features—such as the number of messages sent from i to j in some time interval— and corresponding coefficients. We associate each topic with the interaction pattern that best describes how people interact when talking about that topic.

For each topic $k = 1, \dots, K$:

- Assign topic k to an interaction pattern

$$l_k \sim \text{Uniform}(1, C).$$

Then, we summarize each document's content as a distribution over interaction patterns:

$$\pi_{dc} = \frac{\sum_{k:l_k=c} N_{dk}}{N_d},$$

where N_{dk} is the number of times topic k appears in the document d. In other words, for each document and interaction pattern, we compute the fraction of tokens that were generated using a topic that was assigned to that interaction pattern. We then use this to generate the tie components, which are discussed in the next section.

2.3 Tie Generating Process

The IPTM generates ties and timestamps using a continuous-time process that depends on the interaction patterns' various features and corresponding coefficients. Conditioned on the content generated from

Section 2.1, we assume the following four steps of tie generating process for each document d in a corpus of D documents.

2.3.1 Hypothetical Recipients

We start the recipient generating process by first computing a stochastic intensity for every possible author–recipient pair, combining information about content and network structures.

For each author $i = 1, \ldots, A$, receiver $j = 1, \ldots, A$ $(i \neq j)$ and interaction pattern $c = 1, \ldots, C$, we define the interaction-pattern-specific intensity:

$$\nu_{idjc} = \exp\Big(\boldsymbol{b}_c{}^{\top}\boldsymbol{x}_{idjc}\Big),$$

where $\boldsymbol{b}_c$ is the interaction-pattern-specific coefficients with the prior $\boldsymbol{b}_c \sim N(\boldsymbol{\mu}_b, \Sigma_b)$, and $\boldsymbol{x}_{idjc}$ is the interaction patterns' dynamic network features which vary depending on the hypotheses regarding canonical processes relevant to network theory such as popularity, reciprocity, and transitivity.

We then compute the weighted average of $\{\nu_{idjc}\}_{c=1}^{C}$ and obtain the stochastic intensity—the likelihood of document d being sent from i to j— using the the document's distribution over interaction patterns as mixture weights:

$$\lambda_{idj} = \sum_{c=1}^{C} \pi_{dc}\, \nu_{idjc}.$$

Next, we generate a set of latent recpients for each possible author. In other words, we hypothesize "If i were the author of document d, who would its recipients be?" To do this, we draw each author's set of recipients from a non-empty Gibbs measure (**?**), which is a probability measure we defined in order to prevent from obtaining zero recipient as well as intractible normalizing constants.

For each author $i = 1, \ldots, A$:

- Choose a binary vector $\boldsymbol{u}_{id} = (u_{id1}, \ldots, u_{idA})$

$$\boldsymbol{u}_{id} \sim \text{Gibbs}(\delta, \boldsymbol{\lambda}_{id}),$$

where δ is a real-valued parameter that controls the average number of recipients, with the prior specified as $\delta \sim N(\mu_\delta, \sigma_\delta^2)$, and $\boldsymbol{\lambda}_{id} = (\lambda_{id1}, \ldots \lambda_{idA})$ is the vector of the stochoastic intensities for the author i. In particular, $\text{Gibbs}(\delta, \boldsymbol{\lambda}_{id})$ is defined as

$$p(\boldsymbol{u}_{id}|\delta, \boldsymbol{\lambda}_{id}) \\ = \frac{\exp\Big\{\log\Big(\text{I}(\|\boldsymbol{u}_{id}\|_1 > 0)\Big) + \sum_{j\neq i}\Big(\delta + \log(\lambda_{idj})\Big)u_{idj}\Big\}}{Z(\delta, \boldsymbol{\lambda}_{id})},$$

where $Z(\delta, \boldsymbol{\lambda}_{id})$ is the normalizing constant and $\|\boldsymbol{u}_{id}\|_1$ is the l_1-norm of the binary vector $\boldsymbol{u}_{id}$. Derivation of the normalizing constant is given in Appendix A.

2.3.2 Hypothetical Timestamps

We then generate a hypothetical timestamp for each author–recipient-set pair by saying, "If i were the author of document d and its recipients were $\boldsymbol{u}_{id}$, when would it be sent?" Similar to Section 2.3.1, we define the interaction-pattern-specific rate as below.

For each author $i = 1, \ldots, A$:

$$\xi_{idc} = \boldsymbol{\eta}_c^{\top}\{\boldsymbol{y}_{idjc}\}_{j:u_{idj}=1},$$

where $\boldsymbol{\eta}_c$ is the interaction-pattern-specific coefficients with the prior $\boldsymbol{\eta}_c \sim N(\boldsymbol{\mu}_\eta, \Sigma_\eta)$, and $\boldsymbol{y}_{idjc}$ is the interaction patterns' time-related covariates, which can be any feature that could affect timestamps of the document.

We then calculate the expected value of timestamp as

$$\mu_{id} = \sum_{c=1}^{C} \pi_{dc} g^{-1}(\xi_{idc}),$$

where $g(\cdot)$ is the appropriate link function such as identity, log, or inverse. Again, is the weighted average of $\{\xi_{idc}\}_{c=1}^{C}$ that combines information about content (via $\{\pi_{dc}\}_{c=1}^{C}$) and the time-related covariates.

In modeling the timestamps, we do not assume specific distribution; instead, we provide huge flexibility by following the generalized linear model approach:

$$E(\tau_{id}) = \mu_{id}, \\ V(\tau_{id}) = V(\mu_{id}),$$

where τ_{id} is assumed to be generated from a particular distribution in the exponential family with positive support (*i.e.* $\tau_{id} \in (0, \infty)$) with the mean μ_{id}. Possible choice of distributions include Exponential, Weibull, Gamma, and lognormal[1] distributions, which are commonly used in time-to-event modeling.

2.3.3 Actual Data

Finally, we choose the document's actual author, recipients, and timestamp by selecting the author–recipient-set pair with the earliest timestamp:

$$a_d = \text{argmin}_i(\tau_{id}), \\ \boldsymbol{r}_d = \boldsymbol{u}_{a_d d}, \\ t_d = t_{d-1} + \tau_{a_d d}.$$

[1] For lognormal, take the log-transformation and apply $\mu = E(\log(\tau_{id})) = \mu_{id}$ and $\sigma_\tau^2 = V(\log(\tau_{id})) = V(\mu_{id})$ using identity link function $g = I$. Then, infer the variance parameter σ_τ^2 using $\sigma_\tau^2 \sim \text{Inverse-Gamma}(a_\tau, b_\tau)$.

Therefore, it is an author-driven process in that the author of a document determines its recipients and its timestamp, based on the author's urgency to send the document to chosen recipients.

3 Inference

The generative process is a nice way of describing how a set of documents could theoretically have been generated. However, real documents are not actually generated via this process. As a result, for real-world documents, we only observe the authors $\boldsymbol{a} = \{a_d\}_{d=1}^{D}$, recipients $\boldsymbol{r} = \{\boldsymbol{r}_d\}_{d=1}^{D}$, timestamps $\boldsymbol{t} = \{t_d\}_{d=1}^{D}$ and tokens $\boldsymbol{w} = \{\boldsymbol{w}_d\}_{d=1}^{D}$. On the other hand, the topic-word distributions $\Phi = \{\phi_k\}_{k=1}^{K}$, document-topic distributions $\Theta = \{\boldsymbol{\theta}_d\}_{d=1}^{D}$, topics $\boldsymbol{z} = \{\boldsymbol{z}_d\}_{d=1}^{D}$, topic-interaction pattern assignments $\boldsymbol{l} = \{l_k\}_{k=1}^{K}$, interaction pattern coefficients $\boldsymbol{b} = \{\boldsymbol{b}_c\}_{c=1}^{C}$ and $\boldsymbol{\eta} = \{\boldsymbol{\eta}_c\}_{c=1}^{C}$, hypothetical recipients $\boldsymbol{u} = \{\{\boldsymbol{u}_{id}\}_{i=1}^{A}\}_{d=1}^{D}$, and mean recipient size δ are unobserved. We take a Bayesian approach to infer the latent variables given the observed data.

Our inference goal is to draw samples from the joint posterior distribution

$$\begin{aligned}
&p(\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{b}, \boldsymbol{\eta}, \delta, \boldsymbol{u} | \boldsymbol{w}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}, \alpha, \beta, \boldsymbol{m}, \boldsymbol{\mu}_b, \Sigma_b, \boldsymbol{\mu}_\eta \Sigma_\eta, \mu_\delta, \sigma_\delta^2) \\
&\propto p(\boldsymbol{z}, \boldsymbol{w}, \boldsymbol{l}, \boldsymbol{b}, \boldsymbol{\eta}, \delta, \boldsymbol{u}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t} | \alpha, \beta, \boldsymbol{m}, \boldsymbol{\mu}_b, \Sigma_b, \boldsymbol{\mu}_\eta \Sigma_\eta, \mu_\delta, \sigma_\delta^2) \\
&\propto p(\boldsymbol{z}|\alpha, \boldsymbol{m}) p(\boldsymbol{w}|\boldsymbol{z}, \beta) p(\boldsymbol{l}) p(\boldsymbol{b}|\boldsymbol{\mu}_b, \Sigma_b) p(\boldsymbol{\eta}|\boldsymbol{\mu}_\eta \Sigma_\eta) \\
&\qquad \times p(\delta|\mu_\delta, \sigma_\delta^2) p(\boldsymbol{u}|\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{b}, \delta) p(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}|\boldsymbol{u}, \boldsymbol{z}, \boldsymbol{l}, \boldsymbol{\eta}),
\end{aligned}$$

where Φ and Θ are integrated out using Dirichlet-multinomial conjugacy (**?**). The remaining unobserved variables are sequentially sampled from their joint posterior distribution using Markov chain Monte Carlo (MCMC) methods. Note that we draw the hypothetical recipients $\boldsymbol{u}$ and impute the data by employing data augmentation schemes (**?**). A straightforward Gibbs sampling method are applied for categorical variables $(\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{u})$, while we rely on Metropolis-Hasting for the rest of latent variables that do not have exact conditional posterior distributions.

We omit large part of the sampling equations for the sake of brevity, however, here we illustrate the derivation of the joint posterior distribution of actual data for dth document, $p(a_d, \boldsymbol{r}_d, t_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{b}, \boldsymbol{\eta})$ (see Section 2.3.3), which is the term involved in most of the sampling equations.

$$\begin{aligned}
&p(a_d, \boldsymbol{r}_d, t_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}) \\
&= p(\tau_{a_d d}|\boldsymbol{u}_{a_d d}, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}) \times \prod_{i \neq a_d} p(\tau_{id} > \tau_{a_d d}|\boldsymbol{u}_{id}, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}) \\
&= \varphi_\tau\big(\tau_{a_d d}; \mu_{a_d d}, V(\mu_{a_d d})\big) \\
&\qquad \times \prod_{i \neq a_d} \Big(1 - \Phi_\tau\big(\tau_{a_d d}; \mu_{id}, V(\mu_{id})\big)\Big),
\end{aligned}$$

where φ_τ and Φ_τ are the probability density function (pdf) and cumulative distribution function (cdf) of the specified distribution of timestamps in Section 2.3.2, respecitvely, and $\tau_{a_d d}$ is the observed time-increments $t_d - t_{d-1}$. According to the tie generative process, this joint distribution can be interpreted as '(probability of the observed timestamp generated from the specified distribution of timestamps) × (probability of all hypothetical timestamps greater than the observed time given the specified distribution of timestamps).'

For better performance and interpretability of the topics we infer, we run n_1 iterations of the hyperparameter optimization technique called "new fixed-point iterations using the Digamma recurrence relation" in **?**, for every outer iteration o. In addition, we specify a larger number of inner iterations (n_2, n_3, and n_4) for theose that require Metropolis-Hastings, considering slower mixing. See Appendix B for pseudocode and sampling equations.

4 Applications to Email Network

The IPTM is intended for any network with timestamped, text-valued ties, however, in our application of the model we focus on the analysis of email network, which is the canonical example of dynamic textual communication. Via this application, we demonstrate that the IPTM is effective at predicting and explaining continuous-time textual communications.

4.1 Data

We use a subset of the North Carolina county government email dataset collected by **?** that includes internal email corpora covering the inboxes and outboxes of managerial-level employees of North Carolina county governments. Out of over twenty counties, we chose Dare County, (1) in order to see whether and how communication networks surrounding a notable national emergency—Hurricane Sandy—differed from those surrounding other governmental functions, and (2) to limit the scope of this initial application. The Dare County email network contains around 2,210 emails, sent and received by 27 department managers over a period of 3 months in 2012. To verify that our model is applicable beyond the Dare County email network, we also performed two validation experiments using the Enron email data set (**?**). For this dataset, we took a subset of the original data such that we only include emails between actors who sent over 100 emails, and actors who received over 100 emails from the chosen senders. Emails that were not sent to at least one other active actor were discarded, and also preprocessed to remove any stop words, URLs, quoted text, and signatures. These steps resulted in a total of

3,925 emails involving 33 actors.

4.2 Dynamic Network Features

In Section 2.3.1, we introduced the dynamic network features $\boldsymbol{x}_{idjc}$, which could be flexibly specified according to the researcher's interest. Here, we outline our specifications of the dynamic network statsitics, tailored for the Dare County email network. We follow roughly the same approach as **?**, we employ a suite of eight different effects to be used as the components of $\boldsymbol{x}_{idjc}$—outdegree, indegree, send, receive, 2-send, 2-receive, sibling, and cosibling—to capture common network properties such as popularity, centrality, reciprocity, and transitivity. Visualization of each dynamic network statistics are described in Figure 1, where the upper four features are "dyadic", involving exactly two actors, while the lower four are "triadic", involving exactly three actors.

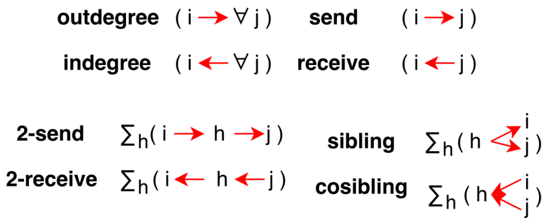


Figure 1: Eight dynamic network statistics used for the Dare County email network and Enron dataset.

Assuming that each network feature has potentially different effects within a number of time intervals (*i.e.* recency effect), we partition the interval $[-\infty, t_d)$ into 4 sub-intervals with equal length in the log-scale, and focus on three time intervals prior to just after the email's timestamp: 4–16 days, 1–4 days, and 0–1 day. We disregard the time interval before 16 days, considering that the Dare County email network only spans 12 weeks in length. We then compute each of the network feature within each time interval to obtain a set of 24 dynamic network features $\boldsymbol{x}_{idjc}$, specific to author i, recipient j, email d, and interaction pattern c. Detailed mathematical formulations and corresponding interpretations of the network statistics are provided in Appendix C.

4.3 Timestamp Specifications

In this section, we exemplify our choice of time-related features that are used to analyze the Dare County email network. First of all, we include the sender and recipient indicators as the component of $\boldsymbol{y}_{idjc}$, since "who is sending to whom" could play a important role in determing "when to send" a document. Taking into account the fact that our data consists of government organizational emails as well as the exploratory results, we added two temporal features into $\boldsymbol{y}_{idjc}$ that stongly affects the timing of documents: the day of the week and time of the day when the previous document was sent.

Moreover, our exploratory analysis revealed that the Dare County email network shows the best fitting when we specify lognormal distribution on the observed timestamps (*i.e.* $\log(\tau_{a_d d}) \sim N(\mu_{a_d d}, V(\mu_{a_d d}) = \sigma_\tau^2))$, while we observed significant lack-of-fit for single parameter distributions such as Exponential distribution (*i.e.* $\tau_{a_d d} \sim \text{Exp}(\mu_{a_d d}))$. Based on this result, we chose lognormal distribution.

5 Experiments

In this section, we conduct a set of posterior predictive experiments using the Dare County email network and the Enron dataset, to showcase the IPTM's predictive performance as compared to alternative modeling approaches.

5.1 Tie Prediction

For a randomly chosen document $d^* \in \{M, M+1, \ldots, D\}$, we fit the IPTM to the corpus consisting of the first $d = \{1, \ldots, d^* - 1\}$ documents, then use the inferred posterior distributions to generate a distribution of predicted tie data $(a_{d^*}, \boldsymbol{r}_{d^*}, t_{d^*})$ conditional on the content in the document $\boldsymbol{w}_{d^*}$, and compare the simulated ones to the observed data. We also compare the IPTM to the alternative model. Several models exist that could be used to model any of these three data types individually, but, to our knowledge, the literature does not offer any models that can be used to jointly generate all three types of tie data integrated into the IPTM. Thus, the alternative model is built upon two separate regression models for the recipients and timestamps, and test if the IPTM has any benefit over other existing models by jointly inferring the parameters that govern the generation of tie data—authors, recipients, and timestamps. Pseudocodes for generating predicted tie data using the IPTM and the regression model are demonstrated in Appendix D.

For both data sets, the Dare County email network and Enron dataset, we randomly selected 200 documents from the later half of the corpus (*i.e.* $M = \frac{D}{2}$) and generated 100 samples of predicted tie data for every document d^*. We ran the same predictive experiments with 21 unique combinations of the number of interaction patterns ($C = 1, 2, 3$) and the number of topics ($K = 2, 5, 10, 25, 50, 75, 100$) as a grid-search based hyperparameter selection process. We compare the predictions in terms of classification accuracy in predicting the authors and recipients, as well as prediction error in the timestamps. Figure 2 presents the F_1 scores on author predictions, multiclass version of the area under the ROC curve (AUC) measure (**?**) on reciptient predictions, and median absolute error (MAE) on timestamp predictions for each document we predicted, all averaged over the entire samples. The outcomes demonstrate the ability IPTM in predicting the author, recipient, and timetsamps of email. **Further comments after we conduct experiment again.**

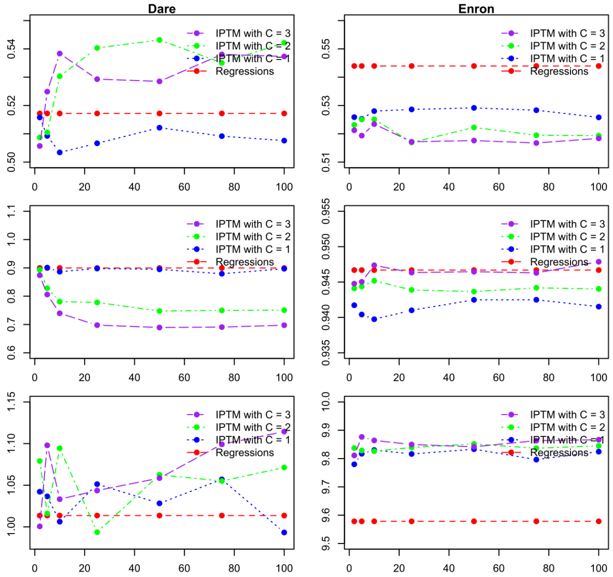


Figure 2: Average AUC, F1 score, MAE: (*top*) Dare County email network. (*bottom*) Enron dataset.

5.2 Topic Coherence

Topic coherence metrics **?** are often used to evaluate the semantic coherence in topic models. In order to test whether the IPTM's incorporation of network features improves the ability of modeling text, we compared the coherence of topics inferred using our model with the coherence of topics inferred using the latent dirichlet allocation (LDA). Instead of re-fit the data using standard LDA algorithms, we used the topic assignments from the IPTM with $C = 1$, which simply makes the IPTM reduced to LDA in terms of topic assignments by unlinking the text and networks. For each model, we varied the number of topics from 1 to 100 and draw five samples from the joint posterior distribution over the latent variable. We evaluated the topics resulting from each sample and averaged over the five samples, where the results are shown in Figure 3. Combined with the results in Section 5.1, this result demonstrates that the IPTM can achieve good predictive performance while producing coherent topics.

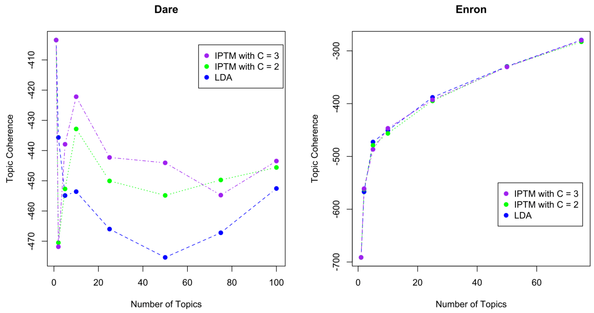


Figure 3: Average topic coherence scores: (*left*) Dare County email network. (*right*) Enron dataset.

5.3 Posterior Predictive Checks

We finally perform posterior predictive checks **?** in order to evaluate the appropriateness of the model specification for the Dare County email network. Formally, we generated entirely new data, by simulating $\{(a_d, \boldsymbol{r}_d, t_d, \boldsymbol{w}_d)\}_{d=1}^{D}$ from the genenerative process in Section 2.1 and 2.3, conditional upon a set of inferred parameter values from the inference in Section 3 (see Appendix D for pseudocode). For the test of goodness-of-fit in terms of network dynamics, we defined multiple network statistics that summarize meaningful aspects of the Dare County email networks: indegree distribution for author activities, outdegree distribution for recipient activities, recipient size distribution, document time-increment distributions, the edgewise shared partner distribution, and the geodesic distance distribution. For content-wise goodness-of-fit, we employed mutual information (MI) in **?**, which is often used to evaluate "bag of words" model assumptions. We then generated 1,000 synthetic networks and texts from the posterior predictive distribution implied by the IPTM and Dare County email network. We applied each discrepancy function to each synthetic network to yield the distributions over the values of the six network statistics and MI. If the model is appropriate, the observed data should not be an outlier with respect to distributions of new data drawn from the posterior predictive distribution.

Figure 4 illustrates the result of posterior predictive checks, showing IPTM's goodness of fit for Dare County data. The results reveal that IPTM captures some important work features of the data, including spreadness and transitivity. **Further comments after we conduct PPC again.**

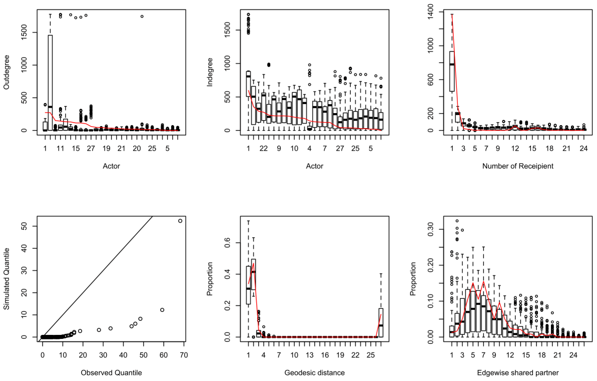


Figure 4: Posterior predictive checks for the Dare County email network: (a) outdegree, (b) indegree, (c) recipient size, (d) QQplot of time-increments, (e) geodesic distance, and (f) edgewise shared partners.

6 Analysis

In order to demonstrate our model's novel ability to identify interaction-pattern-specific communications that exist in both the content and relational structure, we performed an exploratory analysis on the interaction patterns inferred from the Dare County email network using the IPTM. Our main focus was to test three hypotheses: 1) personal or social topics (if any) would exhibit strong reciprocity and transitivity in tie formation, 2) topics about dissemination of information would be characterized by a lack of reciprocity, and 3) topics about Hurricane Sandy would exhibit a very different interaction pattern from the normal day-to-day conversations.

6.1 Topic Assignments

Table 6.1 and 6.1 show top ten words for the five topics that were most strongly associated with interaction pattern 1 and 2, respecitvely. It is pretty clear from the highlighted words that many of the topics in the interaction pattern 1 are about the hurricane. On the contrary, the topics most strongly associated with the interaction pattern 2 are about standard government activities. Very few of their top words are about the hurricane. Together, the assignment of hurricane-related topics to interaction pattern 1 and government-related topics to interaction pattern 2 provide support for our hypothesis that topics about Hurricane Sandy exhibit very a different interaction pattern to other topics.

23	18	11	20	8
change	will	sandy	time	services
order	winds	munis	hours	public
manager	location	hurricane	monday	white
storm	beach	position	leave	director
emergency	hydrant	monday	employees	fyi
coastal	water	point	timesheets	tim
statute	relocation	power	storm	update
evacuation	mirlo	update	employee	status
track	road	storm	tomorrow	board
couple	high	hey	work	approval

21	12	14	4	10
library	marshall	will	board	survey
best	collins	center	meeting	request
start	drive	day	property	call
web	manteo	office	planning	sure
place	phone	great	will	people
visit	box	manager	notice	emergency
albermarle	fax	assistance	january	thought
east	resources	problem	review	seafood
regional	director	lot	commission	mail
system	human	call	list	grant

6.2 Interaction Pattern Coefficients

In theory, we should be able to use the inferred coefficients to understand each interaction pattern's characteristics.

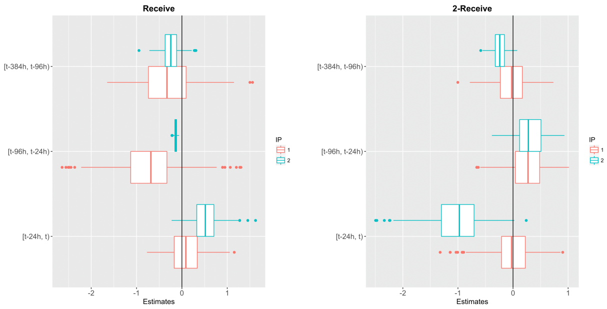


Figure 5: 95% credible intervals of the posterior estimates of $\{\boldsymbol{b}_c\}_{c=1}^{C}$ using Dare County data: (*left*) Recieve. (*right*) 2-Recieve.

7 Conclusions

The IPTM is, to our knowledge, the first model to be capable of jointly modeling the author, recipients, timestamps and contents in time stamped text-valued networks. The IPTM incorporates innovative components, including the modeling of multicast tie formation and the conditioning of ERGM style network generative features on topic-based content. The application to North Carolina county government email data demonstrates, among other capabilities, the effectiveness at the IPTM in separating out both the content and relational structure underlying the normal day-to-day function of an organization and the management of a highly time-sensitive event—Hurricane Sandy. The IPTM is applicable to a variety of networks in which ties are attributed with textual documents. These include, for example, economic sanctions sent between countries and legislation attributed with sponsors and co-sponsors.

Acknowledgements

This work was supported in part by the University of Massachusetts Amherst Center for Intelligent Information Retrieval and in part by National Science Foundation grants DGE-1144860, SES-1619644, and CISE-1320219. Any opinions, findings, and conclusions or recommendations are those of the authors and do not necessarily reflect those of the sponsors.

Appendix

A Normalizing constant of Gibbs measure

In Section 2.3.1, we define the non-empty Gibbs measure such that the probability of author i selecting the binary recipient vector $\boldsymbol{u}_{id}$ is given by

$$\begin{aligned}&P(\boldsymbol{u}_{id}|\delta,\boldsymbol{\lambda}_{id})\\&=\frac{\exp\Big\{\log\Big(\mathrm{I}(\|\boldsymbol{u}_{id}\|_1>0)\Big)+\sum_{j\neq i}\Big(\delta+\log(\lambda_{idj})\Big)u_{idj}\Big\}}{Z(\delta,\boldsymbol{\lambda}_{id})}.\end{aligned}$$

To use this distribution efficiently, we derive a closed-form expression for $Z(\delta,\boldsymbol{\lambda}_{id})$ that does not require brute-force summation over the support of $\boldsymbol{u}_{id}$ (*i.e.* $\forall\boldsymbol{u}_{id}\in[0,1]^A$). We recognize that if $\boldsymbol{u}_{id}$ were drawn via independent Bernoulli distributions in which $P(u_{idj}=1|\delta,\boldsymbol{\lambda}_{id})$ was given by $\mathrm{logit}(\delta+\log(\lambda_{idj}))$, then

$$P(\boldsymbol{u}_{id}|\delta,\boldsymbol{\lambda}_{id})\propto\exp\Big\{\sum_{j\neq i}\Big(\delta+\log(\lambda_{idj})\Big)u_{idj}\Big\}.$$

This is straightforward to verify by looking at

$$P(u_{idj}=1|\boldsymbol{u}_{id[-j]},\delta,\boldsymbol{\lambda}_{id})=\frac{\exp\Big(\delta+\log(\lambda_{idj})\Big)}{\exp\Big(\delta+\log(\lambda_{idj})\Big)+1}.$$

We denote the logistic-Bernoulli normalizing constant as $Z^l(\delta,\boldsymbol{\lambda}_{id})$, which is defined as

$$Z^l(\delta,\boldsymbol{\lambda}_{id})=\sum_{\boldsymbol{u}_{id}\in[0,1]^A}\exp\Big\{\sum_{j\neq i}\Big(\delta+\log(\lambda_{idj})\Big)u_{idj}\Big\}.$$

Now, since

$$\begin{aligned}&\exp\Big\{\log\Big(\mathrm{I}(\|\boldsymbol{u}_{id}\|_1>0)\Big)+\sum_{j\neq i}\Big(\delta+\log(\lambda_{idj})\Big)u_{idj}\Big\}\\&=\exp\Big\{\sum_{j\neq i}\Big(\delta+\log(\lambda_{idj})\Big)u_{idj}\Big\},\end{aligned}$$

except when $\|\boldsymbol{u}_{id}\|_1=0$, we note that

$$\begin{aligned}Z(\delta,\boldsymbol{\lambda}_{id})&=Z^l(\delta,\boldsymbol{\lambda}_{id})-\exp\Big\{\sum_{\forall u_{idj}=0}\Big(\delta+\log(\lambda_{idj})\Big)u_{idj}\Big\}\\&=Z^l(\delta,\log(\lambda_i^{(d)}))-1.\end{aligned}$$

We can therefore derive a closed form expression for $Z(\delta,\boldsymbol{\lambda}_{id})$ via a closed form expression for $Z^l(\delta,\boldsymbol{\lambda}_{id})$. This can be done by looking at the probability of the zero vector under the logistic-Bernoulli model:

$$\begin{aligned}&\frac{\exp\Big\{\sum_{\forall u_{idj}=0}\Big(\delta+\log(\lambda_{idj})\Big)u_{idj}\Big\}}{Z^l(\delta,\boldsymbol{\lambda}_{id})}\\&=\prod_{j\neq i}\Big(1-\frac{\exp\Big(\delta+\log(\lambda_{idj})\Big)}{\exp\Big(\delta+\log(\lambda_{idj})\Big)+1}\Big).\end{aligned}$$

Then, we have

$$Z^l(\delta, \boldsymbol{\lambda}_{id}) \quad = \frac{1}{\prod_{j\neq i} \frac{\exp\{-(\delta+\log(\lambda_{idj}))\}}{\exp\{-(\delta+\log(\lambda_{idj}))\}+1}}.$$

The closed form expression for the normalizing constant under the non-empty Gibbs measure is

$$Z(\delta, \boldsymbol{\lambda}_{id}) = \prod_{j\neq i}\Big(\exp\{\delta + \log(\lambda_{idj})\} + 1\Big) - 1.$$

B MCMC updates

B.1 Psuedocode

Algorithm 1 Markov Chain Monte Carlo updates

```
set initial values
for $o$=1 to $O$ do
    for $n$=1 to $n_1$ do
        optimize $\alpha$ and $\boldsymbol{m}$ using ?
    end
    for $d$=1 to $D$ do
        for $i \in (1:A)_{\backslash a_d}$ do
            for $j \neq i$ do
                draw $\boldsymbol{u}_{idj}$ (Section B.2)
            end
        end
        for $n$=1 to $N_d$ do
            draw $z_{dn}$ (Section B.3)
        end
    end
    for $k$=1 to $K$ do
        draw $l_k$ (Section B.4)
    end
    for $n$=1 to $n_2$ do
        sample $\boldsymbol{b}$ and $\delta$ (Section B.5)
    end
    for $n$=1 to $n_3$ do
        sample $\boldsymbol{\eta}$ (Section B.5)
    end
    for $n$=1 to $n_4$ do
        sample $\sigma_\tau^2$ (Section B.5)
    end
end
```

B.2 Resampling Hypothetical Recipients

First of all, for each document $d = 1, \ldots, D$, we sample the latent recipients for each author other than the observed one (*i.e.* $i \neq a_d$).

For each latent author i, we are going to resample u_{idj}, which is the j^{th} element of the binary vector $\boldsymbol{u}_{id}$, one at a time with random order. The full conditional probability of u_{idj} is:

$$\begin{aligned} p(u_{idj}|\boldsymbol{u}_{id\backslash j}, \boldsymbol{z}, \boldsymbol{l}, \boldsymbol{b}, \delta) &\propto p(u_{idj}, \boldsymbol{u}_{id\backslash j}|\boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{b}, \delta) \\ &\propto p(\boldsymbol{u}_{id}|\delta, \boldsymbol{\lambda}_{id}), \end{aligned}$$

where we use $p(\boldsymbol{u}_{id}|\delta, \boldsymbol{\lambda}_{id})$ defined in Section 2.3.1.

To be more specific, since u_{idj} could be either 0 or 1, we divide into two cases as below:

$$\begin{aligned} &p(u_{idj} = 1|\boldsymbol{u}_{id\backslash j}, \boldsymbol{z}, \boldsymbol{l}, \boldsymbol{b}, \delta) \\ &\propto \frac{\exp\Big\{\log(1) + \sum_{j\neq i}\Big(\delta + \log(\lambda_{idj})\Big)u_{id[j=1]}\Big\}}{Z(\delta, \boldsymbol{\lambda}_{id})} \\ &\propto \exp\Big\{\delta + \log(\lambda_{idj})\Big\}, \end{aligned}$$

where $u_{id[j=1]}$ meaning that the j^{th} element of $\boldsymbol{u}_{id}$ is fixed as 1 (thus making $\log\big(\mathrm{I}(\|\boldsymbol{u}_{id}\|_1 > 0)\big) = 0$). On the other hand,

$$\begin{aligned} &p(u_{idj} = 0|\boldsymbol{u}_{id\backslash j}, \boldsymbol{z}, \boldsymbol{l}, \boldsymbol{b}, \delta) \\ &\propto \frac{\exp\Big\{\log\big(\mathrm{I}(\|\boldsymbol{u}_{id}\|_1 > 0)\big) + \sum_{j\neq i}\Big(\delta + \log(\lambda_{idj})\Big)u_{id[j=0]}\Big\}}{Z(\delta, \boldsymbol{\lambda}_{id})} \\ &\propto \exp\Big\{\log\big(\mathrm{I}(\|\boldsymbol{u}_{id}\|_1 > 0)\big)\Big\}, \end{aligned}$$

where $u_{id[j=0]}$ meaning that the j^{th} element of $\boldsymbol{u}_{id}$ is fixed as 0. In this case, we cannot guarantee $\mathrm{I}(\|\boldsymbol{u}_{id}\|_1 > 0) = 1$, so we have to leave the term. When it is zero, *i.e.* $\mathrm{I}(\|\boldsymbol{u}_{id}\|_1 > 0) = 0$, we will sample 1 with probability 1. From this property of non-empty Gibbs measure, we prevent from the instances where the sender has no recipients to send the document. Now we can use multinomial sampling using the two probabilities above, $p(u_{idj} = 1)$ and $p(u_{idj} = 0)$.

B.3 Resampling Topic Assignments

Next, we resample the topic assignments, one words in a document at a time. Since our content generative process exactly follows that of Latent Dirichlet allocation (**?**), we use the two sampling equations of LDA:

$$\begin{aligned} &p(z_{dn} = k|w_{dn} = w, \boldsymbol{w}_{\backslash dn}, \boldsymbol{z}_{\backslash dn}, \alpha, \boldsymbol{m}) \\ &\propto \frac{N_{dk,\backslash dn} + \alpha m_k}{N_d - 1 + \alpha}, \end{aligned}$$

where the subsscript $\backslash d$ and $\backslash dn$ denote the exclusion of document d and n^{th} element in document d, respectively, and

$$\begin{aligned} &p(w_{dn}|z_{dn} = k, \boldsymbol{w}_{\backslash dn}, \boldsymbol{z}_{\backslash dn}, \beta) \\ &\propto \frac{N_{w_{dn}k,\backslash dn} + \frac{\beta}{V}}{N_{k,\backslash dn} + \beta}, \end{aligned}$$

where $N_{w_{dn}k,\backslash dn}$ is the number of tokens assigned to topic k whose type is the same as that of w_{dn} (excluding w_{dn} itself) and $N_{k,\backslash dn} = \sum_{w=1}^{W} N_{w_{dn}k,\backslash dn}$.

Now, we re-derive the sampling equation reflecting the dynamic network effects as well:

$$
\begin{aligned}
&p(z_{dn}=k|\boldsymbol{z}_{\backslash dn}, \boldsymbol{l}, \boldsymbol{b}, \boldsymbol{\eta}, \delta, \boldsymbol{u}, \boldsymbol{w}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}, \alpha, \beta, \boldsymbol{m}) \\
&\propto p(z_{dn}=k|\boldsymbol{z}_{\backslash dn}, \alpha, \boldsymbol{m}) p(w_{dn}|z_{dn}=k, \boldsymbol{w}_{\backslash dn}, \boldsymbol{z}_{\backslash dn}, \beta) \\
&\quad \times \prod_{i=1}^{A} p(\boldsymbol{u}_{id}|\boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{b}, \delta) p(a_d, \boldsymbol{r}_d, t_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}) \\
&\propto (N_{dk,\backslash dn} + \alpha m_k) \times \frac{N_{w_{dn}k,\backslash dn} + \frac{\beta}{V}}{N_{k,\backslash dn} + \beta} \\
&\times \prod_{i=1}^{A} \frac{\exp\Big\{\log\Big(\mathrm{I}(\|\boldsymbol{u}_{id}\|_1 > 0)\Big) + \sum_{j \neq i} \Big(\delta + \log(\lambda_{idj})\Big) u_{idj}\Big\}}{Z(\delta, \boldsymbol{\lambda}_{id})} \\
&\times \varphi_\tau\big(\tau_{a_d d}; \mu_{a_d d}, V(\mu_{a_d d})\big) \times \prod_{i \neq a_d} \Big(1 - \Phi_\tau\big(\tau_{a_d d}; \mu_{id}, V(\mu_{id})\big)\Big),
\end{aligned}
$$

where $p(\boldsymbol{u}_{id}|\delta, \boldsymbol{\lambda}_{id})$ and $p(\boldsymbol{a}_d, \boldsymbol{r}_d, \boldsymbol{t}_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta})$ are derived in Section 2.3.2 and 2.3.2, respectively. The normalizing constant is not dropped out since any update on z_{dn} may change $\boldsymbol{\lambda}_{id}$ (via π_{dc}), thus $Z(\delta, \boldsymbol{\lambda}_{id})$.

B.4 Resampling Interaction-Patterns

The next variable to resample is the topic-interaction pattern assignments, one topic at a time. We derive the posterior conditional probability for l_k as below:

$$
\begin{aligned}
&p(l_k = c|\boldsymbol{l}_{\backslash k}, \boldsymbol{z}, \boldsymbol{b}, \boldsymbol{\eta}, \delta, \boldsymbol{u}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}) \\
&\propto p(l_k = c) \prod_{d=1}^{D} \prod_{i=1}^{A} p(\boldsymbol{u}_{id}|\boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{b}, \delta) \prod_{d=1}^{D} p(a_d, \boldsymbol{r}_d, t_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}),
\end{aligned}
$$

where $p(l_k = c) = \frac{1}{C}$ so this term disappeas. Thus,

$$
\begin{aligned}
&p(l_k = c|\boldsymbol{l}_{\backslash k}, \boldsymbol{z}, \boldsymbol{b}, \boldsymbol{\eta}, \delta, \boldsymbol{u}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}) \\
&\propto \prod_{d=1}^{D} \prod_{i=1}^{A} \frac{\exp\Big\{\log\Big(\mathrm{I}(\|\boldsymbol{u}_{id}\|_1 > 0)\Big) + \sum_{j \neq i} \Big(\delta + \log(\lambda_{idj})\Big) u_{idj}\Big\}}{Z(\delta, \boldsymbol{\lambda}_{id})} \\
&\quad \times \prod_{d=1}^{D} \Big(\varphi_\tau\big(\tau_{a_d d}; \mu_{a_d d}, V(\mu_{a_d d})\big) \\
&\qquad\qquad \times \prod_{i \neq a_d} \Big(1 - \Phi_\tau\big(\tau_{a_d d}; \mu_{id}, V(\mu_{id})\big)\Big)\Big),
\end{aligned}
$$

where this time we use the product over $d = 1, \ldots, D$ to consider all documents. Again, we maintain the normalizing constant because any update on l_k may change $\boldsymbol{\lambda}_{id}$ (via π_{dc}), thus $Z(\delta, \boldsymbol{\lambda}_{id})$.

B.5 Resampling Coefficients

Lastly, we update the interaction-pattern-specific coefficients, one for recipients $\boldsymbol{b} = \{\boldsymbol{b}_c\}_{c=1}^{C}$ and another for timestamps $\boldsymbol{\eta} = \{\boldsymbol{\eta}_c\}_{c=1}^{C}$. For both, we use the Metropolis-Hastings algorithm with a proposal density Q being the multivariate Normal distribution, with a diagonal covariance matrix multiplied by proposal variance σ_Q^2 and centered on the current values, such that we can cancel out Q-ratio from symmetry.

When we update the recipient coefficients $\boldsymbol{b} = \{\boldsymbol{b}_c\}_{c=1}^{C}$, we jointly sample δ because they share the sampling equations. We accept the new proposed value $\boldsymbol{b}' = \{\boldsymbol{b}'_c\}_{c=1}^{C}$ and δ' with probability equal to:

$$
p(\text{Accept}) = \begin{cases} \frac{p(\boldsymbol{b}', \delta'|\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{\mu}_b, \Sigma_b, \mu_\delta, \sigma_\delta^2)}{p(\boldsymbol{b}, \delta|\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{\mu}_b, \Sigma_b, \mu_\delta, \sigma_\delta^2)} & \text{if} < 1 \\ 1 \quad \text{else} & \end{cases}.
$$

After factorization, we get

$$
\begin{aligned}
&\frac{p(\boldsymbol{b}', \delta'|\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{\mu}_b, \Sigma_b, \mu_\delta, \sigma_\delta^2)}{p(\boldsymbol{b}, \delta|\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{\mu}_b, \Sigma_b, \mu_\delta, \sigma_\delta^2)} \\
&= \frac{p(\boldsymbol{b}'|\boldsymbol{\mu}_b, \Sigma_b) p(\delta'|\mu_\delta, \sigma_\delta^2) \prod_{d=1}^{D} \prod_{i=1}^{A} p(\boldsymbol{u}_{id}|\boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{b}', \delta')}{p(\boldsymbol{b}|\boldsymbol{\mu}_b, \Sigma_b) p(\delta|\mu_\delta, \sigma_\delta^2) \prod_{d=1}^{D} \prod_{i=1}^{A} p(\boldsymbol{u}_{id}|\boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{b}, \delta)},
\end{aligned}
$$

where $p(\boldsymbol{b}|\boldsymbol{\mu}_b, \Sigma_b)$ is the product of $\boldsymbol{b}_c \sim N(\mu_b, \Sigma_b)$ over the interaction patterns $c = 1, \ldots, C$, $p(\delta|\mu_\delta, \sigma_\delta^2)$ is also calculated from $N(\mu_\delta, \sigma_\delta^2)$, and $p(\boldsymbol{u}_{id}|\boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{b}, \delta)$ is the same as what is used in Section B.4.

For the timing coefficients $\boldsymbol{\eta} = \{\boldsymbol{\eta}_c\}_{c=1}^{C}$, we assume lognormal distribution, which we used in our application. Then, the acceptance probability $p(\text{Accept})$ becomes

$$
\begin{aligned}
&\frac{p(\boldsymbol{\eta}'|\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}, \boldsymbol{\mu}_\eta, \Sigma_\eta, \sigma_\tau^2)}{p(\boldsymbol{\eta}|\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}, \boldsymbol{\mu}_\eta, \Sigma_\eta, \sigma_\tau^2)} \\
&= \frac{p(\boldsymbol{\eta}'|\boldsymbol{\mu}_\eta, \Sigma_\eta) \prod_{d=1}^{D} p(a_d, \boldsymbol{r}_d, t_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}', \sigma_\tau^2)}{p(\boldsymbol{\eta}|\boldsymbol{\mu}_\eta, \Sigma_\eta) \prod_{d=1}^{D} p(a_d, \boldsymbol{r}_d, t_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}, \sigma_\tau^2)},
\end{aligned}
$$

where $p(\boldsymbol{\eta}|\boldsymbol{\mu}_\eta, \Sigma_\eta)$ is the product of $\boldsymbol{\eta}_c \sim N(\mu_\eta, \Sigma_\eta)$ over the interaction patterns $c = 1, \ldots, C$, and $p(a_d, \boldsymbol{r}_d, t_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}, \sigma_\tau^2)$ is again the same as what is used in Section B.4.

We also need to sample variance parameter if the specified distribution for timing requires any parameter that controls variance (*e.g.* lognormal or Gamma). Assuming lognormal distribution, we define the variance parameter with σ_τ^2 from the prior $\sigma_\tau^2 \sim \text{Inverse-Gamma}(a_\tau, b_\tau)$. This time we use truncated Normal propoposal ($\sigma_\tau^2 > 0$), thus we cannot cancel out Q-ratio. The acceptance probability $p(\text{Accept})$ is

$$
\begin{aligned}
&\frac{p(\sigma_\tau^{2\prime}|\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}, \boldsymbol{\eta}, a_\tau, b_\tau)}{p(\sigma_\tau^2|\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{u}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}, \boldsymbol{\eta}, a_\tau, b_\tau)} \\
&= \frac{p(\sigma_\tau^{2\prime}|a_\tau, b_\tau) \prod_{d=1}^{D} p(a_d, \boldsymbol{r}_d, t_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}, \sigma_\tau^{2\prime})}{p(\sigma_\tau^2|a_\tau, b_\tau) \prod_{d=1}^{D} p(a_d, \boldsymbol{r}_d, t_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}, \sigma_\tau^2)},
\end{aligned}
$$

where $p(\sigma_\tau^2|a_\tau, b_\tau)$ is calculated from inverse-Gamma density with a_τ and b_τ, and $p(a_d, \boldsymbol{r}_d, t_d|\boldsymbol{u}_d, \boldsymbol{z}_d, \boldsymbol{l}, \boldsymbol{\eta}, \sigma_\tau^2)$ is the same earlier ones.

To avoid computational overflow/underflow, we obtain the log of acceptance ratio by taking the log of $p(\text{Accept})$ above. If the log of a sample from Uniform(0,1) is less than the log-acceptance probability, we accept the proposed values. Otherwise, we reject.

C Specification of Network Features

In Section 4.2, we mentioned that we compute each of the network feature within each time interval,4–16 days, 1–4 days, and 0–1 day ago. Here, we define the degree and dyadic efffects $\boldsymbol{x}_{idjc} = (x^1_{idjc}, x^2_{idjc}, x^3_{idjc})$.

Since we partitioned the interval $[t_d - 16d, t_d)$ into $L = 3$ sub-intervals with equal length in the log-scale, by setting the difference $\Delta_l = (6 \text{ hours}) \times 4^l$ for $l = 1, ..., 3$, we define the intervals I^l_d by

$$\begin{aligned}
&[t_d - 16d, t_d) \\
&= [t_d - 384h, t_d - 96h) \cup [t_d - 96h, t_d - 24h) \cup [t_d - 24h, t_d) \\
&= [t_d - 16d, t_d - 4d) \cup [t_d - 4d, t_d - 1d) \cup [t_d - 1d, t_d) \\
&= I^3_d \cup I^2_d \cup I^1_d,
\end{aligned}$$

where I^l_d is the half-open interval $[t_d - \Delta_l, t_d - \Delta_{l-1})$.

Then, for each time interval $l = 1, \ldots, 3$:

1. $\textbf{outdegree}^l_{id\cdot c} = \sum\limits_{d \in I^l_d} \pi_{dc} I\{i \to \forall j\}$

2. $\textbf{indegree}^l_{\cdot djc} = \sum\limits_{d \in I^l_d} \pi_{dc} I\{\forall i \to j\}$

3. $\textbf{send}^l_{idjc} = \sum\limits_{d \in I^l_d} \pi_{dc} I\{i \to j\}$

4. $\textbf{receive}^l_{idjc} = \sum\limits_{d \in I^l_d} \pi_{dc} I\{j \to i\}$

Next, we define four triadic statistics involving pairs of messages, which are analogous to 2-path statistics commonly used in the network science literature. While **?** adapted full sets of triadic statistics for each combination of time intervals (e.g. $3 \times 3 = 9$), we maintain 3 intervals per each statistic, by defining 3×3 time windows and sum the combination-specific statistics based on the interval where the triads are closed (Refer to Figure 6). As a result, our interval-adjusted definitions of triadic effects become

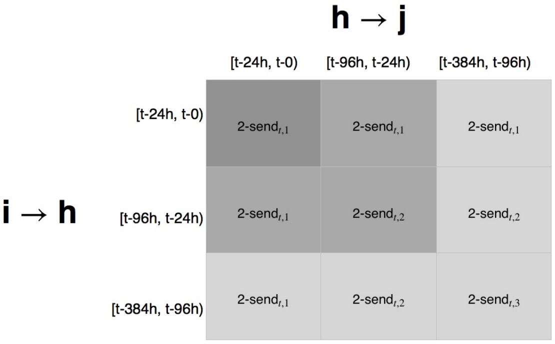
5. $\textbf{2-send}^l_{idjc} =$
$$\sum_{\max(l_1,l_2)=l} \sum_{\substack{h \neq i \\ h \neq j}} \Big(\sum_{d \in I^{l_1}_d} \pi_{dc} I\{i \to h\}\Big)\Big(\sum_{d \in I^{l_2}_d} \pi_{dc} I\{h \to j\}\Big)$$

6. $\textbf{2-receive}^l_{idjc} =$
$$\sum_{\max(l_1,l_2)=l} \sum_{\substack{h \neq i \\ h \neq j}} \Big(\sum_{d \in I^{l_1}_d} \pi_{dc} I\{h \to i\}\Big)\Big(\sum_{d \in I^{l_2}_d} \pi_{dc} I\{j \to h\}\Big)$$

7. $\textbf{sibling}^l_{idjc} =$
$$\sum_{\max(l_1,l_2)=l} \sum_{\substack{h \neq i \\ h \neq j}} \Big(\sum_{d \in I^{l_1}_d} \pi_{dc} I\{h \to i\}\Big)\Big(\sum_{d \in I^{l_2}_d} \pi_{dc} I\{h \to j\}\Big)$$

8. $\textbf{cosibling}^l_{idjc} =$
$$\sum_{\max(l_1,l_2)=l} \sum_{\substack{h \neq i \\ h \neq j}} \Big(\sum_{d \in I^{l_1}_d} \pi_{dc} I\{i \to h\}\Big)\Big(\sum_{d \in I^{l_2}_d} \pi_{dc} I\{j \to h\}\Big)$$

where $l_1 = 1, \ldots, 3$ and $l_2 = 1, \ldots, 3$.



	[t-24h, t-0)	[t-96h, t-24h)	[t-384h, t-96h)
[t-24h, t-0)	2-send$_{\iota,1}$	2-send$_{\iota,1}$	2-send$_{\iota,1}$
[t-96h, t-24h)	2-send$_{\iota,1}$	2-send$_{\iota,2}$	2-send$_{\iota,2}$
[t-384h, t-96h)	2-send$_{\iota,1}$	2-send$_{\iota,2}$	2-send$_{\iota,3}$

Figure 6: 2-send defined for each interval $l = 1, \ldots, 3$. Cells with same shades sum up to one statistic.

D Pseudocode for Experiments

D.1 Tie Prediction

In Section 5.1, we predicted tie data conditional on the content. One important point is that we do not have $\boldsymbol{z}_{d^*}$ and $\boldsymbol{u}_{d^*}$ that came directly from inference on the observed data, as they represent document-specific local latent variables. We need to somehow initialize them conditional on the inferred $\boldsymbol{z}_{1:(d^*-1)}$ and $\boldsymbol{u}_{1:(d^*-1)}$, and update them multiple times such that they are independent of the initialized ones. This can be done via MCMC by iterate 1) drawing the new data, 2) drawing the latent recipients conditional upon the new data, and 3) drawing the topics conditional upon the new data and the latent recipients. We present detailed pseudocode below.

D.2 Prediction using Regressions

We compare the IPTM to the comparative model that is built upon two regression models, which we combine established existing models to make comparable predictions using the same inputs. We train two separate models to use in predicting the recipients and timing of document d^*, respectively. These models are selected to closely mirror the structure of the corresponding components of the IPTM. In each model the documents used for training include $d = 1, \ldots, (d^*-1)$. The model of the recipients is a logistic regression model in which the dependent variable is the observed value of $\boldsymbol{r}_d$. The same network statistics are used as the covariates, with all $c = 1$ (*i.e.* all past interactions are within the single interaction pattern). Specifically, the model used for predicting recipients of document d^* have the form

$$p(u_{id^*j} = 1|\boldsymbol{b}, \delta) = \frac{1}{1 + \exp\big(-(\delta + \boldsymbol{b}^T \boldsymbol{x}_{id^*j})\big)}.$$

The model of timing is via generalized linear model in which the dependent variable is t_d. It is the same approach as the IPTM, except that we fix $c = 1$. In other words, we use

$$E(\tau_{id^*}|\boldsymbol{u}_{d^*}, \boldsymbol{\eta}, \sigma_\tau^2) = g^{-1}(\boldsymbol{\eta}^T \text{GeomMean}(\{\boldsymbol{y}_{id^*j}\}_{j:u_{id^*j=1}})),$$

where $g(\cdot)$ is the appropriate link function. The predicted sender a_{d^*} is determined by simulating τ_{id^*} for each i, and selecting the i that corresponds to the minimum value of τ_{id^*}. The minimum value of τ_{id^*} is the predicted timing for document d^*, which is identical to the IPTM. See below for psesudocode.

D.3 Posterior Predictive Checks (PPC)

To implement posterior predictive checks, we produce a sample from the posterior predictive distribution by drawing new data conditional upon the parameters inferred from entire documents $d = 1, ..., D$ from the posterior distribution of the parameters, repeated over many draws from the posterior distribution.

Formally, we generate new data, which we denote $(\boldsymbol{a}^*, \boldsymbol{r}^*, \boldsymbol{t}^*, \boldsymbol{w}^*)$, conditional upon a set of inferred parameter values from

$$p(\boldsymbol{a}^*, \boldsymbol{r}^*, \boldsymbol{t}^*, \boldsymbol{w}^*|\boldsymbol{z}, \boldsymbol{l}, \boldsymbol{b}, \boldsymbol{\eta}, \delta, \boldsymbol{u}),$$

and the pseudocode is provided below.

Algorithm 2 Predicting tie data for document d^*

```
Inputs:
1. $d^*$, the document to make a prediction
2. $O$, number of outer iterations of inference
3. $R$, the number of inner iterations of prediction
4. $\boldsymbol{w}_{d^*}$, the observed words in the document to predict
5. $p$, the number of burnin iterations
6. $q$, the thinning interval

Run burn-in iterations for inference on $d = 1, \ldots, (d^* - 1)$
for $o$=1 to $O$ do
    run inference on $d = 1, \ldots, (d^* - 1)$
    obtain global variables $(\boldsymbol{l}, \boldsymbol{b}, \boldsymbol{\eta}, \delta)$
    obtain local variables $(\boldsymbol{z}_{1:(d^*-1)}, \boldsymbol{u}_{1:(d^*-1)})$
    initialize $N_{vk}$ and $N_k$ from $\boldsymbol{z}_{1:(d^*-1)}$ and set $N_{d^*k} = 0$
    for $n$ = 1 to $N_{d^*}$ do
        for $k$ = 1 to $K$ do
            $p(z_{d^*n} = k|z_{d^*[1:n-1]}, w_{d^*[1:n]}, \boldsymbol{z}_{1:(d^*-1)}, \boldsymbol{w}_{1:(d^*-1)})$
            $= \frac{N_{d^*k}+\alpha m_k}{n-1+\alpha} \times \frac{N_{w_{d^*n}k}+\frac{\beta}{V}}{N_k+\beta}$
        end
        draw $z_{d^*n}$ from the probability above
        increment $N_{w_{d^*n}z_{d^*n}}$, $N_{z_{d^*n}}$ and $N_{d^*z_{d^*n}}$
    end
    initialize $\boldsymbol{u}_{d^*}$ as below:
    for $i$ = 1 to $A$ do
        sample $d \sim$ Discrete-uniform$(1, d^* - 1)$
        set $\boldsymbol{u}_{id^*} = \boldsymbol{u}_{id}$
        (NOTE: can't directly sample from Gibbs measure due to
        computational complexity)
    end
    for $r$=1 to $p$+$R$ do
        sample $(a_{d^*}, \boldsymbol{r}_{d^*}, t_{d^*})$ following generative process
        for $i \neq a_{d^*}$ do
            for $j \neq i$ do
                Sample $u_{id^*j}$ from Section B.2
            end
        end
        for $n$ = 1 to $N_{d^*}$ do
            Sample $z_{d^*n}$ from Section B.3
        end
        if $r > p$ & $r = qn$ (where $n$ is an integer),
        store $(a_{d^*}, \boldsymbol{r}_{d^*}, t_{d^*})$
    end
end
```

Algorithm 3 Regression predictions for document d^*

```
Input:
1. $\{(a_d, \boldsymbol{r}_d, t_d)\}_{d=1}^{d^*-1}$, Observed documents until $(d^* - 1)$
2. $R$, Number of predictions for document $d^*$

Train models using the two regressions above by Bayesian
estimation using MCMC given the input data.
for $r$ = 1 to $R$ do
    Draw $\boldsymbol{b}, \delta, \boldsymbol{\eta}$ and $\sigma_\tau^2$ (if needed)
    for $i$=1 to $A$ do
        set $\boldsymbol{u}_{id^*}$ to a vector of zeros
        while $\sum\limits_{j \neq i} u_{id^*j} = 0$ do
            for $j \neq i$ do
                Draw $u_{id^*j}$ using $p(u_{id^*j} = 1 | \boldsymbol{b}, \delta)$
            end
        end
        Draw $\tau_{id^*}$ using timing GLM equation
    end
    Set
    $a_{d^*} = \mathrm{argmin}_i(\tau_{id^*})$,
    $\boldsymbol{r}_{d^*} = \boldsymbol{u}_{a_{d^*}d^*}$,
    $t_{d^*} = t_{d^*-1} + \tau_{a_d^* d^*}$.
end
```

Algorithm 4 Generate new data

```
Input:
1. $\boldsymbol{z}$, estimated token topic assignments
2. $\boldsymbol{l}$, estimated topic interaction-pattern assignments
3. $\boldsymbol{b}$, estimated coefficients for recipients
4. $\boldsymbol{\eta}$, estimated coefficients for timestamps
5. $\delta$, estimated recipient size parameter
6. $\boldsymbol{u}$, estimated latent recipients
7. $R$, Number of new data to generate

for $d$ = 1 to $D$ do
    for $r$=1 to $R$ do
        sample ($a_d$, $\boldsymbol{r}_d$, $t_d$) following generative process.
        NOTE:
        One difference from Algorithm 8 is that we initialization.
        Initialize $N_{vk}$ and $N_k$ from the inference on $\boldsymbol{z}$ and observed
        $\mathbf{2}$, instead of zeros.
        store ($a_d$, $\boldsymbol{r}_d$, $t_d$) as $r^{th}$ draw from the posterior
        distribution of the data
    end
end
```