Bayesian Analysis (0000) **00**, Number 0, pp. 1

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

Bomin Kim[*], Aaron Schein[†], Bruce A. Desmarais[‡], and Hanna Wallach[§]

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 timestamped textual communications, according to both the network dynamics that they reflect and their content. To define the IPTM, we integrate the hyperedge event model (HEM)—a generative model for events that can be represented as directed edges with one sender and one or more receivers or one receiver and one or more senders—and latent Dirichlet allocation (LDA)—a generative model for topic-based content. The IPTM assigns each document to an "interaction pattern"—a generative process for contents and ties that is governed by a topic distribution and a set of dynamic network features. 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.

MSC 2010 subject classifications: Primary 60K35, 60K35; secondary 60K35.

Keywords: dynamic network model, topic modeling, text analysis, email data analysis.

1 Introduction

In recent decades, real-time digitized textual communication has developed into a ubiquitous form of social and professional interaction (Kanungo and Jain, 2008; Szóstek, 2011; Burgess et al., 2004; Pew, 2016). 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 (LDA) (Blei et al., 2003) to incorporate link data as well as textual content have been developed recently (McCallum et al., 2005; Lim et al., 2013; Krafft et al., 2012). 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—such as the exponential random graph model (ERGM) (Robins et al., 2007; Chatterjee et al., 2013; Hunter et al., 2008). The ERGM is the canonical model for modeling the structure of a static network. It is

[*] Department of Statistics, Pennsylvania State University bzk147@psu.edu
[†] College of Information and Computer Sciences, UMass Amherst aschein@cs.umass.edu
[‡] Department of Political Science, Pennsylvania State University bdesmarais@psu.edu
[§] Microsoft Research NYC hanna@dirichlet.net

DOI: 0000

flexible enough to specify a generative model that accounts for nearly any pattern of tie formation (e.g., reciprocity, clustering, popularity effects) (Desmarais and Cranmer, 2017). Several models have been developed that handle time-stamped ties in which tie formation is governed by structural dynamics similar to those used in ERGMs (Perry and Wolfe, 2013; Butts, 2008; Snijders, 1996). We develop the interaction-partitioned topic model (IPTM) which simultaneously models the network structural patterns that govern time-stamped tie formation, and the content in the communications.

The models on which we build, including the relational event model (Butts, 2008), the point process model (Perry and Wolfe, 2013), and most closely the hyperedge event model (HEM), provide frameworks for explaining or predicting ties between nodes using the network sub-structures in which the two nodes are embedded (e.g., predict a tie is highly likely to form between two nodes if those two nodes have many shared partners). Models based on network structure 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 these network 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 (Bratton and Rouse, 2011; Alemán and Calvo, 2013); the study of alliance networks in which networks reflect countries' co-signing of treaties, but exclude treaty text (Camber Warren, 2010; Cranmer et al., 2012b,a; Kinne, 2016); the study of scientific co-authorship networks that exclude the text of the co-authored papers (Kronegger et al., 2011; Liang, 2015; Fahmy and Young, 2016); and the study of text-based interaction on social media (e.g., users tied via 'mentions' on twitter) (Yoon and Park, 2014; Peng et al., 2016; Lai et al., 2017).

In defining and testing the IPTM we embed core conceptual property—interaction pattern—to 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 (Section 2). The IPTM leads to an efficient inference using the Markov Chain Monte Carlo (MCMC) algorithm (Section 3) and acheives good predictive peformance (Section 4). Finally, the IPTM discovers interesting and interpretable latent structure through application to email corpora of internal communications by government officials in Dare County, NC (Section 4).

2 Interaction-partitioned Topic Model

In this section we define the IPTM by describing a process according to which documents are generated in continuous time. Data generated under the IPTM consists of D unique documents. A single document, indexed by $d \in [D]$, is represented by the four components: the sender $a_d \in [A]$, an indicator vector of receivers $\boldsymbol{r}_d = \{u_{dr}\}_{r=1}^{A}$, the timestamp $t_d \in (0, \infty)$, and a set of tokens $\boldsymbol{w}_d = \{w_{dn}\}_{n=1}^{N_d}$ that comprise the text of the document, where N_d denotes the total number of tokens in a document. For simplicity, we assume that documents are ordered by time such that $t_d < t_{d+1}$.

2.1 Interaction Patterns

They key idea that combines the IPTM component modeling "what" with the component modeling "who," "whom," and "when" is that different documents comes from the introduction of "interaction patterns." Each interaction pattern $c \in [C]$ is characterized by a set of features that affects networks and timestamps—such as the number of messages sent from a to r in some time interval—and corresponding coefficients. We associate each document with the interaction pattern that best describes how people interact, and that is reflected to what people talk about via topic assignments. To be specific, each document $d \in [D]$ draws an interaction pattern c_d as below

$$c_d \sim \text{Categorical}(\frac{\psi_1}{\sum_c \psi_c}, \dots, \frac{\psi_C}{\sum_c \psi_c}), \tag{2.1}$$

and we assume the interaction-pattern specific weights from Gamma distribution

$$\psi_c \sim \Gamma(\frac{\gamma_0}{C}, \zeta), \tag{2.2}$$

where γ_0 and ζ are the shape and rate parameters.

2.2 Generative Process for Contents

The words $\boldsymbol{w}_d$ are generated by extending the well-known Bayesian topic model, latent Dirichlet allocation (LDA) (Blei et al., 2003) to follow the form of Bayesian Poisson Tucker decomposition (BPTD) (Schein et al., 2016). As in LDA, we generate the corpus-wide global variables that describe the content via topics. First, the topic-word factors for each topic $k \in [K]$ and word $v \in [V]$ are also gamma-distributed; however, we assume that these factors are drawn directly from an uninformative gamma prior

$$\phi_{kv} \sim \Gamma(\epsilon_0, \epsilon_0). \tag{2.3}$$

Next, following the cluster-based topic model, document d has the document-topic distribution

$$\boldsymbol{\theta}_d \sim \text{Dirichlet}(\alpha, \boldsymbol{m}_{c_d}), \tag{2.4}$$

where α are the concentration parameter and $\boldsymbol{m} = (m_1, \dots, m_K)$ is the base measure. In order to capture the overall prevalence of each topic in the corpus, we assume that each $\boldsymbol{m}_c$ is given Dirichlet priors with a single corpus-level base measure $\boldsymbol{m}$

$$\boldsymbol{m}_c \sim \text{Dirichlet}\Big(\alpha_1, \boldsymbol{m}\Big), \tag{2.5}$$

where α_1 is the concentration parameter determining the extent to which the group-specific base measures are affected by the corpus-level base measure. Finally, the corpus-level base measure is assumed to have Dirichlet prior with uniform base measure

$$\boldsymbol{m} \sim \text{Dirichlet}\Big(\alpha_0, (\frac{1}{K}, \dots, \frac{1}{K})\Big). \tag{2.6}$$

Given that $\bar{N}_d = \max(1, N_d)$ where N_d is known, a topic z_{dn} is drawn from the document-topic distribution and then a word w_{dn} is drawn from the chosen topic for each $n \in [\bar{N}_d]$—i.e.,

$$\begin{aligned} z_{dn} &\sim \text{Multinomial}(\boldsymbol{\theta}_d), \\ w_{dn} &\sim \text{Multinomial}(\phi_{z_{dn}}). \end{aligned} \tag{2.7}$$

2.3 Generative Process for Network Dynamics

The generative process for network portion of the documents—the senders, receivers, and timestamps—exactly follows that of the hyperedge event model (HEM). While the HEM can be applied for two types of hyperedges—edges with (1) one sender and one or more receivers, and (2) one or more senders and one receiver—here we only present the generative process for those involving one sender and one or more receivers, considering our email applications. One notable feature of the IPTM generative process is that we draw auxiliary variables that serve as candidate data. Data is generated from the IPTM through a sampling process applied to the auxiliary variables. The auxiliary variables drawn for document d include, for each sender a, a time increment from document $d-1$ at which sender a would send document d, and an $A-1$ length vector indicating which nodes would receive document d if it were sent by sender a. The data generated for document d under the IPTM corresponds to the sender that would send document d the soonest—at the smallest time increment from document $d-1$. The receivers of document d generated under the IPTM correspond to those receivers to which the sender with the minimum time increment would have sent document d. We explain these steps in more detail below.

Candidate receivers

For every possible sender–receiver pair $(a, r)_{a \neq r}$, we define the "receiver intensity"—an approximate logit of the probability that edge d is being sent from sender a to receiver r—as a linear combination of statistics relevant to the receiver selection process:

$$\lambda_{adr} = \boldsymbol{b}^\top \boldsymbol{x}_{adr}, \tag{2.8}$$

where $\boldsymbol{b}$ is a P–dimensional vector of coefficients and $\boldsymbol{x}_{adr}$ is a set of receiver selection features. Such features can capture common network processes that include, but are not limited to, popularity, reciprocity, and transitivity, as well as the effects of attribute features defined on the sender, receivers, and/or sender-receiver pairs (e.g., the gender of the sender and/or receiver, an indicator of whether the sender is a supervisor of the receiver's). In addition, we include an intercept term to account for the average (or baseline) number of receivers. We place a Normal prior $\boldsymbol{b} \sim N(\boldsymbol{\mu}_b, \Sigma_b)$.

A basic assumption of the HEM is that, at any given time, everyone is planning to send a document to one or more receivers. So for each edge d and each sender a, the HEM draws a list of candidate receivers that would be receivers on edge d if a were the sender. For an edge d, we first define an $A \times A$ matrix $\boldsymbol{u}_d$ where the a^{th} row denotes sender a's receiver vector of zeros and 1's—i.e., 1's indicate the nodes to which

a intends to send edge d. We then assume that each receiver vector $\boldsymbol{u}_{ad}$ comes from a modification of the multivariate Bernoulli (MB) distribution (Dai et al., 2013)—a model that has been used to model graphs in which the state of each edge indicator is drawn independently from an edge-specific Bernoulli distribution. In order to avoid drawing edges with no receivers, we define a probability measure "MB$_G$" motivated by the Gibbs measure (Fellows and Handcock, 2017). The probability measure we define amounts to a non-empty Gibbs measure, in which the all-zero vector is excluded from the support of the multivariate Bernoulli distribution. As a result, this measure helps us to 1) allow a sender to select multiple receivers for a single edge, 2) force the sender to select at least one receiver, and 3) ensure a tractable normalizing constant for the receiver selection distribution. To be specific, we draw a binary vector $\boldsymbol{u}_{ad} = (u_{ad1}, \ldots, u_{adA})$

$$\boldsymbol{u}_{ad} \sim \mathrm{MB}_G(\boldsymbol{\lambda}_{ad}), \tag{2.9}$$

where $\boldsymbol{\lambda}_{ad} = \{\lambda_{adr}\}_{r=1}^{A}$. In particular, we define $\mathrm{MB}_G(\boldsymbol{\lambda}_{ad})$ as

$$\Pr(\boldsymbol{u}_{ad}|\boldsymbol{b}, \boldsymbol{x}_{ad}) = \frac{1}{Z(\boldsymbol{\lambda}_{ad})} \exp\Big(\log\big(\mathrm{I}(\|\boldsymbol{u}_{ad}\|_1 > 0)\big) + \sum_{r \neq a} \lambda_{adr} u_{adr}\Big), \tag{2.10}$$

where $Z(\boldsymbol{\lambda}_{ad}) = \prod_{r\neq a} \big(\exp(\lambda_{adr}) + 1\big) - 1$ is the normalizing constant and $\|\cdot\|_1$ is the l_1–norm. Again, this is approximately equivalent to assuming that each u_{adr} is drawn with the probability of 1 being $\mathrm{logit}(\lambda_{adr})$, excluding the case when $u_{adr} = 0$ for all $r \in [A]$. We provide detailed derivation steps for the normalizing constant $Z(\boldsymbol{\lambda}_{ad})$ in Appendix B.

Candidate timestamps

Similarly, for each sender and edge combination, the HEM draws a candidate time at which the edge would be sent. The timing rate for sender a and edge d is

$$\mu_{ad} = g^{-1}(\boldsymbol{\eta}^\top \boldsymbol{y}_{ad}), \tag{2.11}$$

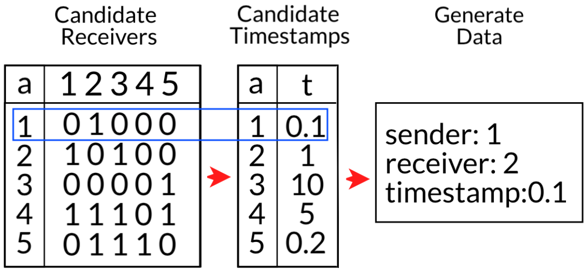
where $\boldsymbol{\eta}$ is a Q–dimensional vector of coefficients with a Normal prior $\boldsymbol{\eta} \sim N(\boldsymbol{\mu}_\eta, \Sigma_\eta)$, $\boldsymbol{y}_{ad}$ is a set of event timing features—covariates that could affect timestamps of the edge, and $g(\cdot)$ is the appropriate link function such as identity, log, or inverse.

In modeling "when," we do not directly model the timestamp t_d. Instead, we assume that each sender's "time increment"—i.e., waiting time to next interaction since t_{d-1}—is drawn from a specific distribution in the exponential family. We define the time increment from edge $d-1$ to edge d as τ_d (i.e., $\tau_d = t_d - t_{d-1}$) and specify the distribution of candidate timestamps with sender-specfic mean μ_{ad}. Following the generalized linear model (GLM) framework (Nelder and Baker, 1972), we assume the mean and variance of the τ_{ad} satistify

$$\begin{aligned} E(\tau_{ad}) &= \mu_{ad}, \\ V(\tau_{ad}) &= V(\mu_{ad}), \end{aligned} \tag{2.12}$$

where τ_{ad} here is a positive real number. Possible choices of distribution include exponential, Weibull, gamma, and log-normal[1] distributions, which are commonly used in

[1] The log-normal distribution is not exponential family but can be used via modeling of $\log(\tau_d)$.



a	1	2	3	4	5
1	0	1	0	0	0
2	1	0	1	0	0
3	0	0	0	0	1
4	1	1	1	0	1
5	0	1	1	1	0

a	t
1	0.1
2	1
3	10
4	5
5	0.2

sender: 1
receiver: 2
timestamp:0.1

Figure 1: An illustrative example of the generative process of the HEM.

time-to-event modeling (Rao, 2000; Rizopoulos, 2012). Based on the specific distribution, we may need other latent variables to draw the time increment, to account for the variance of time increments, beyond the coefficients for the features used to model the rate. $V(\mu)$—e.g., the shape parameter k for the Weibull, the shape parameter θ for the gamma, and the variance parameter σ_τ^2 for the log-normal. We use $f_\tau(\cdot;\mu,V(\mu))$ and $F_\tau(\cdot;\mu,V(\mu))$ to denote the probability density function (p.d.f) and cumulative density function (c.d.f), respectively, with mean μ and variance $V(\mu)$.

Senders, receivers, and timestamps

Finally, under the HEM the observed sender, receivers, and timestamp of edge d is generated by selecting the sender–receiver-set pair with the smallest time increment (Snijders, 1996):

$$\begin{aligned} a_d &= \text{argmin}_a(\tau_{ad}),\\ \boldsymbol{r}_d &= \boldsymbol{u}_{a_d d},\\ t_d &= t_{d-1} + \tau_{a_d d}. \end{aligned} \tag{2.13}$$

Therefore, it is a sender-driven process in that the receivers and timestamps of an edge are jointly determined by the sender's urgency to send the edge to the selected receivers. Note that our generative process accounts for tied events such that in case of tied events (i.e., multiple senders draw exactly same candidate timestamps), we observe all of the tied events without assigning the orders of tied events. Algorithm 1 summarizes the entire generative process for directed edges with one sender and one or more receivers, and Figure 1 presents an illustrative example on how the dth edge is generated, assuming $t_{d-1} = 0$ for simplicity.

3 Posterior inference

In this section we describe how we invert the generative process to obtain the posterior distribution over the latent variables—candidate receivers $\{\boldsymbol{u}_d\}_{d=1}^D$, coefficients for edge covariates $\boldsymbol{b}$, and coefficients for timestamp covariates $\boldsymbol{\eta}$—conditioned on the observed data $\{(a_d, \boldsymbol{r}_d, t_d)\}_{d=1}^D$, covariates $\{(\boldsymbol{x}_d, \boldsymbol{y}_d)\}_{d=1}^D$, and hyperparamters $(\boldsymbol{\mu}_b, \Sigma_b, \boldsymbol{\mu}_\eta, \Sigma_\eta)$. We draw the samples using Markov chain Monte Carlo (MCMC) methods, repeat-

Algorithm 1 Generative Process: one sender and one or more receivers

```
Input: number of edges and nodes $(D, A)$, covariates $(\boldsymbol{x}, \boldsymbol{y})$, and the coefficients $(\boldsymbol{b}, \boldsymbol{\eta})$

for d=1 to D do
  for a=1 to $A$ do
    for r=1 to $A$ (r $\neq$ a) do
      set $\lambda_{adr} = \boldsymbol{b}^\top \boldsymbol{x}_{adr}$
    end for
    draw $\boldsymbol{u}_{ad} \sim \text{MB}_G(\boldsymbol{\lambda}_{ad})$
    set $\mu_{ad} = g^{-1}(\boldsymbol{\eta}^\top \boldsymbol{y}_{ad})$
    draw $\tau_{ad} \sim f_\tau(\mu_{ad}, V(\mu_{ad}))$
  end for
  if $n \geq 2$ tied events then
    set $a_d, \dots, a_{d+n-1} = \text{argmin}_a(\tau_{ad})$
    set $\boldsymbol{r}_d = \boldsymbol{u}_{a_d d}, \dots, \boldsymbol{r}_{d+n-1} = \boldsymbol{u}_{a_{d+n-1} d}$
    set $t_d, \dots, t_{d+n-1} = t_{d-1} + \min_a \tau_{ad}$
    jump to $d = d + n$
  else
    set $a_d = \text{argmin}_a(\tau_{ad})$
    set $\boldsymbol{r}_d = \boldsymbol{u}_{a_d d}$
    set $t_d = t_{d-1} + \min_a \tau_{ad}$
  end if
end for
```

edly resampling the value of each latent variable from its conditional posterior via a Metropolis-within-Gibbs sampling algorithm. In this section, we provide each latent variable's conditional posterior, and demonstrate how we perform the prior-posterior simulator test of Geweke (2004), in which we evaluate the integrity of both our mathematical derivations and the sampling code, for the HEM. At the end of this section, we provide the pseudocode of our MCMC algorithm in Algorithm 2.

3.1 Conditional posteriors

Candidate receivers

In the HEM, direct computation of the posterior densities for the latent variables $\boldsymbol{b}$ and $\boldsymbol{\eta}$—i.e., $P(\boldsymbol{b}|\boldsymbol{x}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t})$ and $P(\boldsymbol{\eta}|\boldsymbol{y}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t})$—is not possible. However, it is possible to augment the data by candidate receivers $\boldsymbol{u}$ such that we can obtain their conditional posterior by collapsing the known distributions—$P(\boldsymbol{b}, \boldsymbol{u}|\boldsymbol{x}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t})$ and $P(\boldsymbol{\eta}, \boldsymbol{u}|\boldsymbol{y}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t})$—through integrating out $\boldsymbol{u}$. We adapt this common tool in Bayesian statistics called "data augmentation" (Tanner and Wong, 1987; Neal and Kypraios, 2015). Since u_{adr} is a binary random variable, its new value may be sampled directly from a multinomial

Algorithm 2 MCMC Algorithm

```
Input: number of outer and inner iterations $(O, I_1, I_2)$ and initial values of $(\boldsymbol{u}, \boldsymbol{b}, \boldsymbol{\eta})$

for o=1 to O do
  for d=1 to D do
    for a = 1 to A do
      for r = 1 to A (r ≠ a) do
        update $u_{adr}$ using Gibbs update —equation (3.1)
      end for
    end for
  end for
  for n=1 to $I_1$ do
    update $\boldsymbol{b}$ using M-H algorithm—equation (3.2)
  end for
  for n=1 to $I_2$ do
    update $\boldsymbol{\eta}$ using M-H algorithm—equation (3.3) or (3.4)
  end for
  if extra parameter for $V(\mu)$ then
    update the variance parameter using M-H algorithm—equation (3.3) or (3.4)
  end if
end for
summarize the results with the last chain of $\boldsymbol{b}$ and $\boldsymbol{\eta}$
```

distribution with probabilities

$$\begin{aligned}\Pr(u_{adr} = 1 | \boldsymbol{u}_{ad\backslash r}, \boldsymbol{b}, \boldsymbol{x}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}) &\propto \exp(\lambda_{adr});\\ \Pr(u_{adr} = 0 | \boldsymbol{u}_{ad\backslash r}, \boldsymbol{b}, \boldsymbol{x}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}) &\propto \mathrm{I}(\|\boldsymbol{u}_{ad\backslash r}\|_1 > 0),\end{aligned} \tag{3.1}$$

where $I(\cdot)$ is the indicator function that is used to prevent from the instances where a sender chooses zero number of receivers.

Coefficients for edge covariates

Unlike the candidate receivers above, new values for $\boldsymbol{b}$ cannot be sampled directly from its conditional posterior, but may instead be obtained using the Metropolis–Hastings (M-H) algorithm. Assuming an uninformative prior (i.e., $N(0, \infty)$), the conditional posterior over $\boldsymbol{b}$ is

$$\Pr(\boldsymbol{b} | \boldsymbol{u}, \boldsymbol{x}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}) \propto \prod_{d=1}^{D} \prod_{a=1}^{A} \frac{1}{Z(\boldsymbol{\lambda}_{ad})} \exp\Big(\log\big(\mathrm{I}(\|\boldsymbol{u}_{ad}\|_1 > 0)\big) + \sum_{r \neq a} \lambda_{adr} u_{adr}\Big). \tag{3.2}$$

Coefficients for timestamp covariates

Likewise, we use the M-H algorithm to update the latent variable $\boldsymbol{\eta}$. Assuming an uninformative prior $\boldsymbol{\eta}$ (i.e., $N(0, \infty)$), the conditional posterior for an untied event case is

$$\Pr(\boldsymbol{\eta}|\boldsymbol{u}, \boldsymbol{y}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}) \propto \prod_{d=1}^{D} \Big(f_\tau(\tau_d; \mu_{a_d d}, V(\mu_{a_d d})) \times \prod_{a \neq a_d} \big(1 - F_\tau(\tau_d; \mu_{ad}, V(\mu_{a_d d}))\big) \Big), \quad (3.3)$$

where $f_\tau(\tau_d; \mu_{a_d d}, V(\mu_{a_d d}))$ is the probability that the d^{th} observed time increment comes from the specified distribution $f_\tau(\cdot)$ with the observed sender's mean $\mu_{a_d d}$, and $\prod_{a \neq a_d} \big(1 - F_\tau(\tau_d; \mu_{ad}, V(\mu_{a_d d}))\big)$ is the probability that the rest of (unobserved) senders for event d all draw time increments greater than τ_d. Moreover, under the existence of tied events, the conditional posterior of $\boldsymbol{\eta}$ is written as

$$\begin{aligned} \Pr(\boldsymbol{\eta}|\boldsymbol{u}, \boldsymbol{y}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t}) \propto \prod_{m=1}^{M} \Big(& \prod_{d : t_d = t_m^*} f_\tau(t_m^* - t_{m-1}^*; \mu_{a_d d}, V(\mu_{a_d d})) \\ & \times \prod_{a \notin \{a_d\}_{d : t_d = t_m^*}} \big(1 - F_\tau(t_m^* - t_{m-1}^*; \mu_{ad}, V(\mu_{a_d d}))\big) \Big), \end{aligned} \quad (3.4)$$

where $t_1^*, \ldots, t_M^*$ are the unique timepoints across D events ($M \leq D$). If $M = D$ (i.e., no tied events), equation (3.4) reduces to equation (3.3). Note that when we have the latent variable to quantify the variance in time increments $V(\mu)$ (based on the choice of timestamp distribution in Section **??**), we also use equation (3.3) (or equation (3.4) in case there exist tied events) for the additional M-H update—e.g., $\Pr(k|\boldsymbol{\eta}, \boldsymbol{u}, \boldsymbol{y}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t})$ for Weibull, $\Pr(\theta|\boldsymbol{\eta}, \boldsymbol{u}, \boldsymbol{y}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t})$ for gamma, and $\Pr(\sigma_\tau^2|\boldsymbol{\eta}, \boldsymbol{u}, \boldsymbol{y}, \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{t})$ for log-normal distribution.

4 Application to email data

We now present a case study applying our method to Montgomery county government email data. Our data come from the North Carolina county government email dataset collected by ben Aaron et al. (2017) 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 Montgomery County to 1) test our model using data with a large proportion of hyperedges (16.76%), all of which are emails sent from one sender to two or more receivers, and 2) limit the scope of this initial application. The Montgomery County email network contains 680 emails, sent and received by 18 department managers over a period of 3 months (March–May) in 2012. For this case study, we formulate a HEM specification through definitions of the edge covariates $\boldsymbol{x}$ and timestamp covariates $\boldsymbol{y}$. We then report a suite of experiments—out-of-sample prediction for model selection and posterior predictive checks—that illustrate how alternative formulations of the HEM can be compared, and evaluate how well the HEM recovers the distribution of the observed data. Finally, we demonstrate an exploratory analysis of Montgomery County email data using the model estimates to discover substantively meaningful patterns in organizational communication networks.

4.1 Covariates

Edge covariates

A primary purpose of any network model is to use the posterior distributions to learn which features predict and/or explain edge formation (e.g., is edge formation reciprocal, are edges more likely to be formed among nodes with the same gender). This email application specifically give rise to the following question: "To what extent are nodal, dyadic or triadic network effects relevant to predicting future emails?" As an illustrative example, we form the receiver selection features $\boldsymbol{x}$ for Montgomery County email data using nodal, dyadic, and triadic covariates. First, as we want to test whether gender plays a role in receiver selection process, we include three nodal covariates—the gender information of sender and receiver, and their homophily indicator (i.e., an indicator of whether the sender and receiver are of the same gender). Additionally, we include four interval-based nodal network covariates—outdegree of sender (i.e., the number of edges sent), indegree of receiver (the number of edges received), hyperedge size of sender (i.e., the number of total receivers of edges from the sender), and the interaction between (i.e., scalar product of) outdegree and hyperedge size—to study the effect of nodal behaviors on future interactions. For dyadic and triadic network effects, we employ the network statistics in Perry and Wolfe (2013) and summarize past interaction behaviors based on the time interval prior to and including t_{d-1}. Specifically, our time interval tracks 7 days prior to the last email was sent $l_d = (t_{d-1} - 7 \text{ days}, t_{d-1}]$. For $a \in [A], r \in [A]$, and $d \in [D]$, we define 14 covariates for $\boldsymbol{x}_{adr}$:

1. intercept: $x_{adr1} = 1$;
2. gender of sender: $x_{adr2} = I(\text{gender of sender } a = \text{female})$;
3. gender of receiver: $x_{adr3} = I(\text{gender of receiver } r = \text{female})$;
4. gender homophily: $x_{adr4} = I(x_{adr2} = x_{adr3})$;
5. outdegree of sender: $x_{adr5} = \sum_{d':t_{d'} \in l_d} I(a_{d'} = a)$;
6. indegree of receiver: $x_{adr6} = \sum_{d':t_{d'} \in l_d} I(u_{d'r} = 1)$;
7. hyperedge size of sender: $x_{adr7} = \sum_{d':t_{d'} \in l_d} \sum_{r=1}^{A} I(a_{d'} = a)\, I(u_{d'r} = 1)$;
8. interaction between outdegree and hyperedge size: $x_{adr8} = x_{adr5} \times x_{adr7}$;
9. send: $x_{adr9} = \sum_{d':t_{d'} \in l_d} I(a_{d'} = a)\, I(u_{d'r} = 1)$;
10. receive: $x_{adr10} = \text{send}(r, a)$;
11. 2-send: $x_{adr11} = \sum_{h \neq a,r} \text{send}(a, h)\, \text{send}(h, r)$;

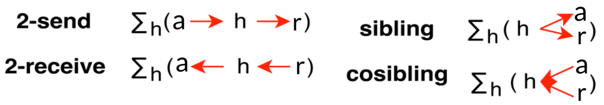


Figure 2: Visualization of triadic statistics: 2-send, 2-receive, sibling, and cosibling.

12. 2-receive: $x_{adr12} = \sum_{h \neq a,r} \text{send}(h,a)\,\text{send}(r,h)$;
13. sibling: $x_{adr13} = \sum_{h \neq a,r} \text{send}(h,a)\,\text{send}(h,r)$;
14. cosibling: $x_{adr14} = \sum_{h \neq a,r} \text{send}(a,h)\,\text{send}(r,h)$;

where $I(\cdot)$ is an indicator function. The network statistics (5–14) are designed so that their coefficients have a straightforward interpretations. The statistics "outdegree" and "indegree" measure the gregariousness and popularity effects of the node by counting the number of emails sent from a and received by r, respectively, within the last 7 days. The gregariousness effect refers to the tendency for nodes that sent many edges in the past to continue to do so in the future. The popularity effect refers to the tendency for nodes that received many edges in the past to continue to do so in the future. Moreover, in order to capture individual tendency of send emails to two or more receivers, we include the statistic "hyperedge size"—the number of emails sent from a within last 7 days where emails with k number of receivers are counted as k separate emails—as a variant of outdegree statistic, accounting for hyperedges. We also include the interaction term between outdegree and hyperedge size. This interaction allows us to model a possible tradeoff between the hyperedge size and the total number of edges sent. Dyadic statistics "send" and "receive" are defined as above such that these covariates measure the number of emails sent from a to r and r to a, respectively, within the last 7 days. In the example of triadic statistics, the covariate "2-send" counts the pairs of emails involving some node h distinct from a and r such that emails from a to h and h to r are both observed within the last 7 days. The 2-send statistic captures the tendency for emails to close transitive triads (i.e., triads in which a sends to r and h, and r sends to h). We include other triadic covariates that behave similarly and exhibit analogous interpretations, which are illustrated in Figure 2.

Timestamp covariates

For the event timing features $\boldsymbol{y}$ introduced in Section **??**, we identify a set of covariates which may effect "time to the next email." Similar to the edge covariates, we include nodal statistics which are time-invariant (such as gender or manager status) or time-dependent (such as the network statistics used for $\boldsymbol{x}$). In addition, we select some edge-specific covariates based on the temporal aspect of the $(d-1)^{th}$ email—e.g., whether the previous email was sent (1) during the weekend and (2) before or past midday (AM/PM)—since we expect the email interactions within county government to be less active during the weekend and in the evening. To be specific, the timestamp statistics are defined as

1. intercept: $y_{ad1} = 1$;
2. gender of sender: $y_{ad2} = I(\text{gender of sender } a = \text{female})$;
3. manager status of sender: $y_{ad3} = I(\text{sender } a \text{ is the County Manager})$;
4. outdegree of sender: $y_{ad4} = \sum_{d':t_{d'} \in l_d} I(a_{d'} = a)$;
5. indegree of sender: $y_{ad5} = \sum_{d':t_{d'} \in l_d} I(u_{d'a} = 1)$;
6. weekend indicator of previous email: $y_{ad6} = I(t_{d-1} \text{ is during the weekend})$;
7. PM indicator of previous email: $y_{ad7} = I(t_{d-1} \text{ in PM})$.

Note that our generative process for timestamps in Section **??** is sender-oriented where the sender deterimes when to send the email, thus we incorporate network statistics that depends on a only—outdegree of sender a and indegree of sender a.

4.2 Model selection

The HEM has many component parts that need to be specified by the user (i.e., the receiver selection features $\boldsymbol{x}$, the selection of the event timing features $\boldsymbol{y}$, and the distribution of time increments f). Many of these components will be specified based on user expertise (e.g., regarding which features would drive receiver selection), but some decisions may require a data-driven approach to model specification. For example, though theoretical considerations may inform the specification of features, subject-matter expertise is unlikely to inform the decision regarding the family of the event time distribution. Furthermore, since different distribution families (and model specifications more generally) may involve different size parameter spaces, any data-driven approach to model comparison must guard against over-fitting the data. In this section we present a general-purpose approach to evaluating the HEM specification using out-of-sample prediction. We illustrate this approach by comparing alternative distributional families for the event timing component of the model. Here, we specifically compare the predictive performance from two distributions—log-normal and exponential. We particularly choose the log-normal distribution based on some exploratory analysis (e.g., histogram and simple regressions) on raw time increments data, and take exponential distribution as an alternative since exponential is the most commonly specified distribution for time-to-event data, and is also used in the stochastic actor-oriented models (SAOMs) (Snijders, 1996) as well as their extensions (Snijders et al., 2007).

We evaluate the model's ability to predict edges and timestamps from Montgomery County email data, conditioned on their "training" part of the data. To perform the experiment, we separately formed a test split of each three model components—sender, receivers, and timestamps—by randomly selecting "test" data with probability $p = 0.10$, and setting the test data to missing. Any missing variables were imputed by drawing samples from their conditional posterior distributions, given the observed data, model estimates, and current values of imputed test data. We then run inference to update

Algorithm 3 Out-of-Sample Predictions

```
Input: data $\{(a_d, \boldsymbol{r}_d, t_d)\}_{d=1}^D$, number of new data to generate $R$, and initial values of $(\boldsymbol{b}, \boldsymbol{\eta}, \boldsymbol{u}, \sigma_\tau^2)$

Test splits (with $p = 0.1$):
draw test senders (out of $D$ senders)
draw test receivers (out of $D \times (A-1)$ receiver indicators $\{\{\boldsymbol{r}_{dr}\}_{r\in[A]_{\backslash a_d}}\}_{d=1}^D$)
draw test timestamps (out of $D$ timestamps)
set the test data as "missing" (NA)

Imputation and inference:
for $r = 1$ to $R$ do
  for $d = 1$ to $D$ do
    if $a_d = \text{NA}$ then
      for $a = 1$ to $A$ do
        compute $\pi_a = P(a_d = a|\cdot)$ using equation (3.3) (without the product term)
      end for
      draw $a_d \sim \text{multinomial}(\pi_a)$
    end if
    for $r \in [A]_{\backslash a_d}$ do
      if $r_{dr} = \text{NA}$ then
        draw $r_{dr}$ from multinomial with probability $P(r_{dr} = 1|\cdot)$ and $P(r_{dr} = 0|\cdot)$
        using equation (3.1)
      end if
    end for
    if $t_d = \text{NA}$ then
      draw $\boldsymbol{\tau}_d^{new}$ from equation (3.3) (without the product term) via importance
      sampling[2]
    end if
    run inference and update $(\boldsymbol{u}, \boldsymbol{b}, \boldsymbol{\eta})$ given the imputed and observed data
  end for
  store the estimates for test data
end for
```

the latent variables given the imputed and observed data. We iterate the two steps—imputation and inference—multiple times to obtain posterior samples of the held out test data. Algorithm 3 outlines this procedure in detail. We run the experiment and measure the predictive performance of two separate time distributions using $N = 500$ predicted samples, by comparing the predictions in terms of classification accuracy in predicting the senders and receivers, as well as prediction error in the timestamps. We summarize the results of prediction experiments for missing senders, receivers, and timestamps in Figure 3. First, we compare the posterior probability of correct senders for each of the missing emails $\{d : a_d = \text{NA}\}$, which corresponds to $\pi_{a_d} = P(a_d = a_d^{obs}|\cdot)$ in Algorithm 3. We call this measure the "correct sender posterior probability." On Figure 3 (a), we draw boxplots for the distribution of mean correct sender posterior

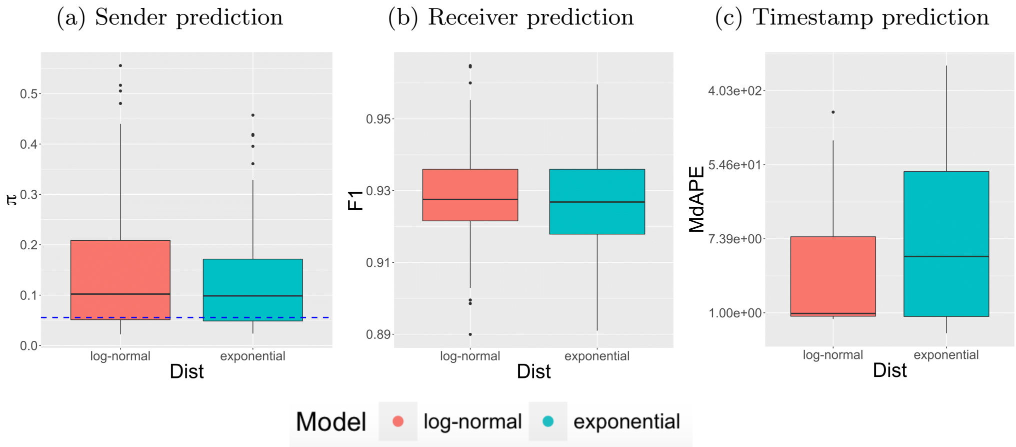

(a) Sender prediction (b) Receiver prediction (c) Timestamp prediction

Figure 3: Comparison of predictive performance between log-normal and exponential distributions: (a) correct sender posterior probability from sender predictions, (b) F_1 scores from receiver predictions, and (c) median absolute relative error from timestamp predictions. Blue line in (a) represents the correct sender probability expected by random guess—i.e., $1/A = 1/18 \approx 0.056$.

probability—i.e., $\hat{\pi}_{a_d} = \frac{1}{N}\sum_{n=1}^{N} \pi_{a_d}^{n}$—across the missing emails. The results show that both log-normal and exponential distributions acheive better predictive performance for missing senders compared to what is expected under random guess (i.e., choose one out of A possible senders $= 1/18$), with log-normal distribution showing higher performance than exponential distribution. Secondly, since the receiver vector is binary, we compute F_1 scores for missing receiver indicators (i.e., all d and r with r_{dr}=NA) by taking the harmonic mean of precision and recall:

$$\begin{aligned} F_1 &= 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}, \text{ where} \\ \text{recall} &= \frac{\text{TP}}{\text{TP+FN}} \text{ and precision} = \frac{\text{TP}}{\text{TP+FP}}, \end{aligned} \tag{4.1}$$

with TP denoting true positive (i.e., $\boldsymbol{r}_{dr}^{obs} = \boldsymbol{r}_{dr}^{pred} = 1$), FN denoting false negative (i.e., $\boldsymbol{r}_{dr}^{obs} = 1$ but $\boldsymbol{r}_{dr}^{pred} = 0$), and FP denoting false positive (i.e., $\boldsymbol{r}_{dr}^{obs} = 0$ but $\boldsymbol{r}_{dr}^{pred} = 1$). Although the generative process for edges (Section **??**) is not directly affected by the choice of timestamp distribution, Figure 3 (b) reveals slight difference between log-normal and exponential in their performance in predicting missing receiver indicators, where log-normal on average outperforms exponential. Finally, the prediction error for d^{th} missing timestamp is estimated using the median of absolute relative errors[3] across $N = 500$ predictions:

[3] Hyndman and Koehler (2006) refer to this as median absolute percentage error (MdAPE).

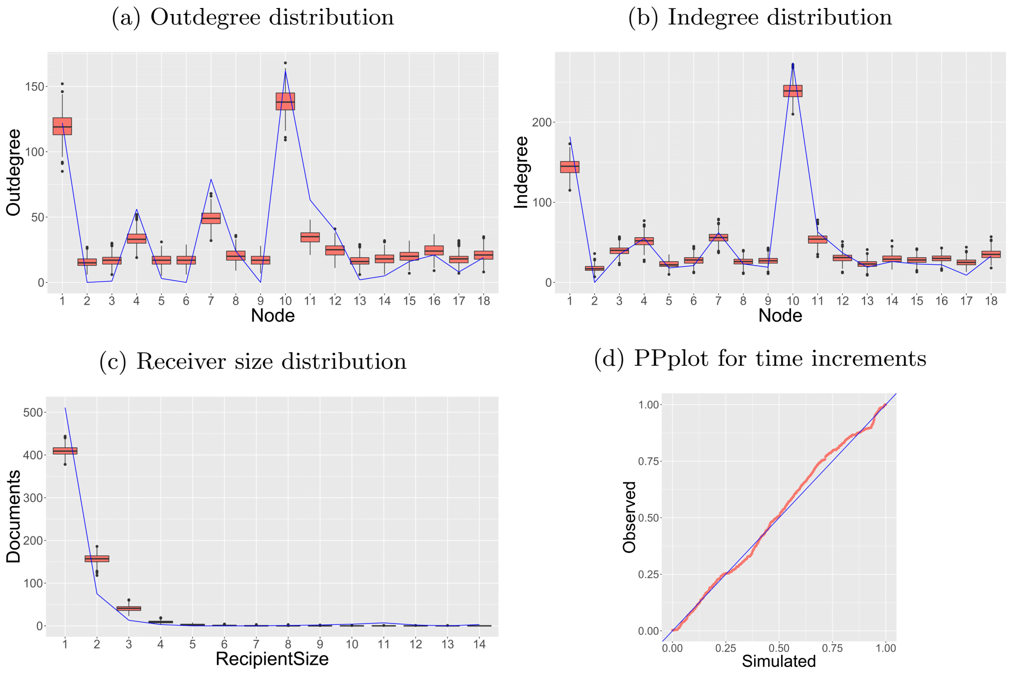
$$e_{\tau_d} = \text{median}\Bigg(\bigg|\frac{\tau_d^{obs} - \tau_d^{pred_1}}{\tau_d^{obs}}\bigg|, \ldots, \bigg|\frac{\tau_d^{obs} - \tau_d^{pred_N}}{\tau_d^{obs}}\bigg|\Bigg). \tag{4.2}$$

Figure 3 (c) presents boxplots for the median absolute relative errors. where plot the estimates in a log-scale. Surprisingly, we have a huge benefit in the performance of timestamp prediction when we assume log-normal distribution for time-increments compared to exponential distribution. This difference can be explained by overdispersion in exponential distribution, because there exists greater variability in the time increments of emails than would be expected under exponential distribution. As illustrated above, we can use this out-of-sample prediction task for two uses—1) to provide an effective answer to the question "how does the HEM perform at filling in the missing components of time-stamped network data?" and 2) to offer one standard way to determine the distribution of time increments in Section **??**.

4.3 Posterior predictive checks

In this section, we perform posterior predictive checks (PPC) (Rubin et al., 1984) to evaluate the appropriateness of our model specification for Montgomery County email data. We formally generated entirely new data by simulating $N = 500$ synthetic email datasets $\{(a_d, \boldsymbol{r}_d, t_d)\}_{d=1}^{D}$ from the generative process in Section 2, conditional upon a set of inferred latent variables from the inference in Section 4.4. For the test of goodness-of-fit in terms of network dynamics, we use multiple statistics that summarize meaningful aspects of the data: outdegree distribution—the number of edges sent by each node, indegree distribution—the number of edges received by each node, receiver size distribution—the number of receivers on each edge, and a probability-probability (PP) plot for time increments.

Figure 4 illustrates the results of poterior predictive checks using the log-normal distribution, which shows better performance in Section 4.2. The upper two plots show node-specific posterior predictive degree distributions across $N = 500$ synthetic samples, where the left one for outdegree statistic and the right plot is for indegree statistic. For both plots, the x-axis represents the nodes ($A = 1, \ldots, 18$), and the y-axis represents the number of emails sent or received by the node. When compared with the observed outdegree and indegree statistics (red lines), our model recovers the overall distribution of sending and receiving activities across the nodes. For example, node 1 and 10 show significantly higher level of both sending and receiving activities relative to the rest, and the model-simulated data captures those big jumps, showing acceptable fit to the data. Outdegree distribution of some low-activity nodes are not precisely recovered, however, indegree distribution looks much better. Since we use more information in the receiver selection process (i.e., network effects) while we rely solely on minimum time increments when choosing the observed sender, these results are expected. The lower left plot is the distribution of receiver sizes, where the x-axis spans over the size of receivers 1 to 14 (which is the maximum size of observed receivers) and the y-axis denotes the number of emails with x-number of receivers. The result shows that our model is underestimating emails with one receiver while overestimating emails with two, three, and four receivers. One explanation behind what we observe is that the


(a) Outdegree distribution

(b) Indegree distribution

(c) Receiver size distribution

(d) PPplot for time increments

Figure 4: PPC results from log-normal distribution. Blue lines denote the observed statistics in (a)–(c) and denotes the diagonal line in (d).

model is trying to recover so-called “broadcast” emails, which are the emails with ≥ 10 number of receivers, so that the intercept estimate b_1 is slightly moved toward right. It would be an interesting problem in future research to consider how the hyperedge size distribution can be further modified to capture this distribution more accurately. The plot on the lower right is the PP plot for time increments, which depicts the two cumulative distribution functions—one for simulated time increments and another for observed time increments—against each other in order to assess how closely two data sets agree. Here, the closeness to the diagonal line connecting $(0, 0)$ and $(1, 1)$ gives a measure of difference between the simulated and observed time increments, and our PP plot shows that we have great performance in reproducing the observed time distribution. Our findings from the predictive experiments in Section 4.2 are further revealed in the PPC from exponential distribution, where the PPC plots comparing log-normal and exponential distributions are presented in Appendix C.

4.4 Exploratory analysis

Based on the prediction experiments in Section 4.2, we interpret the results from the HEM using the log-normal distribution. We assume weakly informative priors for latent variables such as $\boldsymbol{b} \sim N(\boldsymbol{\mu}_b = \mathbf{0}, \Sigma_b = 2 \times I_P)$, $\boldsymbol{\eta} \sim N(\boldsymbol{\mu}_\eta = \mathbf{0}, \Sigma_\eta = 2 \times I_Q)$, and

$\sigma_\tau^2 \sim$ inverse-Gamma$(a = 2, b = 1)$, and run MCMC algorithm in Algorithm 2 with $O = 55,000$ outer iterations with a burn-in of 15,000, where we thin by keeping every 40th sample. While the inner iterations for σ_τ^2 is fixed as 1, we specify the inner iterations $I_1 = 20$ for $\boldsymbol{b}$ and $I_2 = 10$ for $\boldsymbol{\eta}$ to adjust for slower convergence rates. Convergence diagnostics including the traceplots and Geweke diagnostics (Geweke et al., 1991) are provided in Appendix D.

Coefficients for edge covariates

Figure 5 shows the boxplots summarizing posterior samples of $\boldsymbol{b}$, where Figure 5 (a) displays the coefficients for nodal covariates and 5 (b) displays the coefficients for dyadic and triadic covariates. Since we use the logit functional form

$$\text{logit}(\lambda_{adr}) = \log\left(\frac{\lambda_{adr}}{1-\lambda_{adr}}\right) = b_1 + b_2 x_{adr2} \ldots + b_{14} x_{adr14},$$

and can interpret the $\boldsymbol{b}$ estimates in terms of odds ratios $\frac{\lambda_{adr}}{1-\lambda_{adr}} = \exp(b_1 + b_2 x_{adr2} \ldots + b_{14} x_{adr14})$. We find the effects of nodal coavariates "gender of sender" and "gender of receiver" are both nearly always negative in the posterior samples. The log odds that any other node will be added as a receiver of an email is approximately 0.5 less if the sender is a woman. The posterior distribution of the statistic "outdegree" is mostly negative, if sender a sent n number of emails to anyone last week, then sender a is approximately $\exp(-0.109 \times n) \approx (0.897)^n$ times less likely to send an email to r. However, this straightforward interpretation of the outdegree statistic only applies when the hyperedge size is low. The scenario in which a sender sends a lot of low-hyperedge-size emails may arise due to the use of email for a one-on-one conversation. The large positive estimates of the interaction between hyperedge size and outdegree indicate that those who have recently sent many emails with many receivers on each email are likely to continue sending these "broadcast" style emails. This scenario may arise from someone being responsible for distributing timely announcements. When we look at the effect of "indegree," we see a clear popularity effect—those who have received a lot of emails a lot recently are likely to continue receiving a lot of emails. If the receiver r received n number of emails over the last week, sender a is $\exp(0.086 \times n) \approx (1.091)^n$ times more likely to send an email to r.

When we look at the effects of dyadic and triadic covariates, one thing that stands out is the large and positive posterior distribution of the statistic"send" (i.e., number of times sender a sent emails to receiver r over the last week) with the posterior mean $\hat{b}_9 = 0.274$, implying that if sender a sent n number of emails to r last week, then sender a is approximately $\exp(0.274 \times n) \approx (1.315)^n$ times more likely to send an email to r. The posterior distributions for the reciprocity effect (receive), and the four triadic effects, are all fairly evenly spread around zero, so our results do not justify conclusions regarding the nature of these effects in the Montgomery county email network.

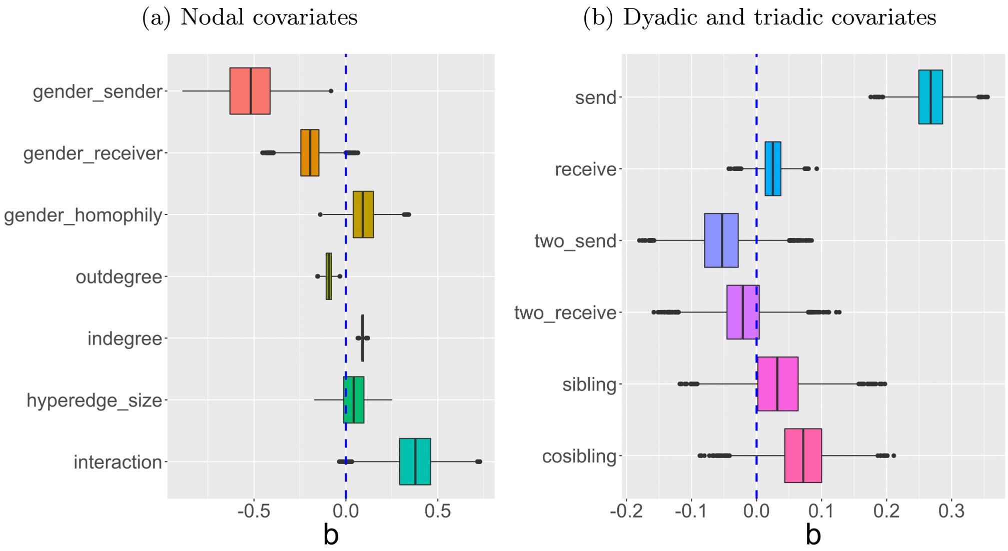


Figure 5: Posterior distribution of $\boldsymbol{b}$ estimates.

Coefficients for timestamp covariates

For timestamp covariates, Figure 6 shows the boxplots summarizing posterior samples of $\boldsymbol{\eta}$. Note that interpretations of the estimated coefficients for $\hat{\boldsymbol{\eta}}$ should be based on the specified timeunit of the datset, which we use "hour" for Montgomery county email data. Moreover, since we assume log-normal distribution for time increments, the coefficients are interpreted in terms of the change in the average log time.

$$\log(\tau_{ad}) \sim N(\mu_{ad}, \sigma_\tau^2), \text{ with}$$
$$\mu_{ad} = \eta_1 + \eta_2 y_{ad2} \ldots + \eta_7 y_{ad7}.$$

The posterior estimates of two temporal effects—"weekend" and "PM"—indicate that if the $(d-1)^{th}$ email was sent during the weekend or after midday, then the time to d^{th} email is expected to take $\exp(1.552) \approx 4.722$ hours and $\exp(0.980) \approx 2.665$ hours longer, respectively, compared to their counterparts (i.e., weekdays and am). On the contrary, the covariates "manager", "outdegree", and "indegree" shorten the amount of time to next email. For example, being a county manager (i.e., the lead county administrator) lowers the expected value of $\log(\tau_{ad})$ by $\hat{\eta}_3 = -1.070$, since the manager in general sends or receives a lot more emails which may shorten the response time and many of those emails. The posterior estimates for the "outdegree" and "indegree" statistics, where the posterior means are approximately $\hat{\eta}_4 = -0.206$ and $\hat{\eta}_5 = -0.060$, respectively. These effects indicate that those who are involved in either sending or receiving a lot of emails recently are likely to send emails with greater speed. The posterior distribution for the effect of the gender of the manager is pretty evenly spread around zero. In addition, the posterior mean estimates for variance parameter σ_τ^2 in the lognormal distribution

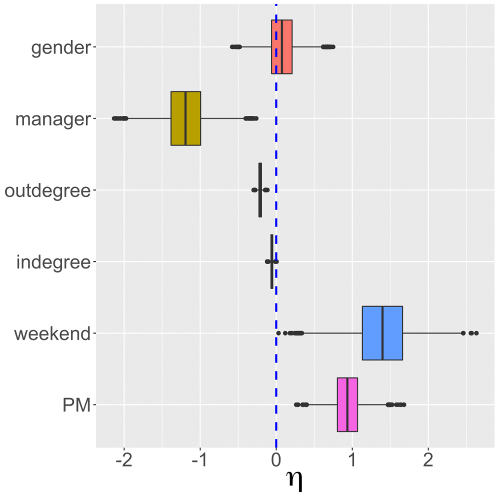


Figure 6: Posterior distribution of $\boldsymbol{\eta}$ estimates.

is approximately $\hat{\sigma}_{\tau}^2 = 14.093$ with its 95% credible interval $(12.709, 15.555)$, indicating that there exists large variability in the time increments of emails.

5 Conclusion

Motivated by a growing class of dynamic network models which deal with edges exchanged in continuous time, the hyperedge event model (HEM) can effectively learn the underlying dynamics in edge and timestamp formations, providing novel insights to the literature. The HEM explicitly models hyperedges through a receiver-selection distribution that forces the sender to select at least one recevier, which is a more realistic approach for events with one sender and one or more receivers and one or more sender and one receiver compared to treating them as pure duplicates. In modeling the timestamps (more precisely time increments) of events, our generalized linear model (GLM) based formulation offers new innovations by eliminating the need to stick with one parameter distribution (e.g., exponential distribution). To our knowledge, the HEM is the only existing model that can be used to generate the sender, recipients, and time stamp of interactions in real time. To make better use of the proposed model, we provide an algorithm for predictive experiments that help to learn which specification of HEM provides a better fit to the data.

We have demonstrated the effectiveness of our model by analyzing Montgomery County government emails, where emails serve as a canonical example of directed edges with one sender and one or more receivers. The estimated effects for edge covariates reveal that the HEM is able to understand the structural dynamics similar to those used in the exponential random graph model (ERGM), but we also learn about the effects of timestamp covariates by integrating a survival model for event timing. Although we illustrate the entire framework and application in the context of one type of hyperedge, one sender and one or more receivers, our model can be easily extended to allow the opposite case, one or more sender and one receiver, by slight modification of the generative process (shown in Appendix A). This extension involves promising applications to socio-political networks such as international sanctions and co-sponsorship of bills, and biological networks such as those formed through neural dendrites.

Acknowledgments

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

Appendix A: Alternative generative process

Algorithm 4 Generative Process: one receiver and one or more senders

```
Input: number of edges and nodes $(D, A)$, covariates $(\boldsymbol{x}, \boldsymbol{y})$, and the coefficients $(\boldsymbol{b}, \boldsymbol{\eta})$

for d=1 to D do
  for r=1 to $A$ do
    for a=1 to $A$ (a $\neq$ r) do
      set $\lambda_{adr} = \boldsymbol{b}^\top \boldsymbol{x}_{adr}$
    end for
    draw $\boldsymbol{u}_{rd} \sim \mathrm{MB}_G(\boldsymbol{\lambda}_{rd})$
    set $\mu_{rd} = g^{-1}(\boldsymbol{\eta}^\top \boldsymbol{y}_{rd})$
    draw $\tau_{rd} \sim f_\tau(\mu_{rd}, \sigma_\tau^2)$
  end for
  if $n \geq 2$ tied events then
    set $r_d = \mathrm{argmin}_r(\tau_{rd})$
    set $\boldsymbol{a}_d = \boldsymbol{u}_{r_d d}, \ldots, \boldsymbol{a}_{d+n-1} = \boldsymbol{u}_{r_{d+n-1} d}$
    set $t_d, \ldots, t_{d+n-1} = t_{d-1} + \min_r \tau_{rd}$
    jump to $d = d + n$
  else
    set $r_d = \mathrm{argmin}_r(\tau_{rd})$
    set $\boldsymbol{a}_d = \boldsymbol{u}_{r_d d}$
    set $t_d = t_{d-1} + \min_r \tau_{rd}$
  end if
end for
```

Appendix B: Normalizing constant of MB$_G$

Our probability measure "MB$_G$"—the multivariate Bernoulli distribution with non-empty Gibbs measure—defines the probability of sender a selecting the binary receiver vector $\boldsymbol{u}_{ad}$ as

$$\Pr(\boldsymbol{u}_{ad}|\boldsymbol{b}, \boldsymbol{x}_{ad}) = \frac{1}{Z(\boldsymbol{\lambda}_{ad})} \exp\Big(\log\big(\mathrm{I}(\|\boldsymbol{u}_{ad}\|_1 > 0)\big) + \sum_{r \neq a} \lambda_{adr} u_{adr}\Big),$$

where the receiver intensity is a linear combination of edge covariates—i.e., $\lambda_{adr} = \boldsymbol{b}^\top \boldsymbol{x}_{adr}$—as defined in Secton **??**.

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

$$\Pr(\boldsymbol{u}_{ad}|\boldsymbol{b}, \boldsymbol{x}_{ad}) \propto \exp\Big(\sum_{r \neq a} \lambda_{adr} u_{adr}\Big).$$

This is straightforward to verify by looking at

$$\Pr(u_{adr} = 1|\boldsymbol{u}_{ad[-r]}, \boldsymbol{b}, \boldsymbol{x}_{ad}) = \frac{\exp(\lambda_{adr})}{\exp(\lambda_{adr}) + 1}.$$

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

$$Z^l(\boldsymbol{\lambda}_{ad}) = \sum_{\boldsymbol{u}_{ad} \in [0,1]^A} \exp\Big(\sum_{r \neq a} \lambda_{adr} u_{adr}\Big).$$

Now, since

$$\exp\Big(\log\Big(\mathrm{I}(\|\boldsymbol{u}_{ad}\|_1 > 0)\Big) + \sum_{r \neq a} \lambda_{adr} u_{adr}\Big) = \exp\Big(\sum_{r \neq a} \lambda_{adr} u_{adr}\Big),$$

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

$$\begin{aligned} Z(\boldsymbol{\lambda}_{ad}) &= Z^l(\boldsymbol{\lambda}_{ad}) - \exp\Big(\sum_{\forall u_{adr} = 0} \lambda_{adr} u_{adr}\Big) \\ &= Z^l(\boldsymbol{\lambda}_{ad}) - 1. \end{aligned}$$

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

$$\frac{1}{Z^l(\boldsymbol{\lambda}_{ad})} \exp\Big(\sum_{\forall u_{adr} = 0} \lambda_{adr} u_{adr}\Big) = \prod_{r \neq a} \Big(1 - \frac{\exp(\lambda_{adr})}{\exp(\lambda_{adr}) + 1}\Big).$$

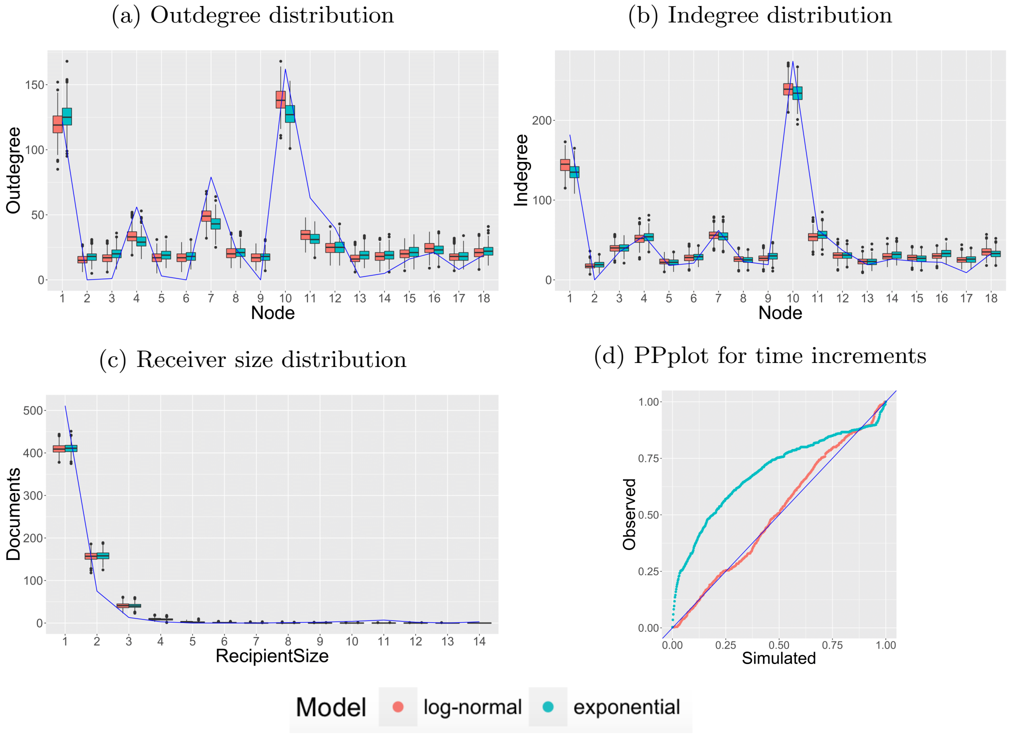
Then, we have

$$\frac{1}{Z^l(\boldsymbol{\lambda}_{ad})} \quad = \prod_{r\neq a} \frac{1}{\exp(\lambda_{adr})+1}.$$

Finally, the closed form expression for the normalizing constant is

$$Z(\boldsymbol{\lambda}_{ad}) = \prod_{r\neq a} \big(\exp(\lambda_{adr})+1\big) - 1.$$

Appendix C: Comparison of PPC results: log-normal vs. exponential


(a) Outdegree distribution

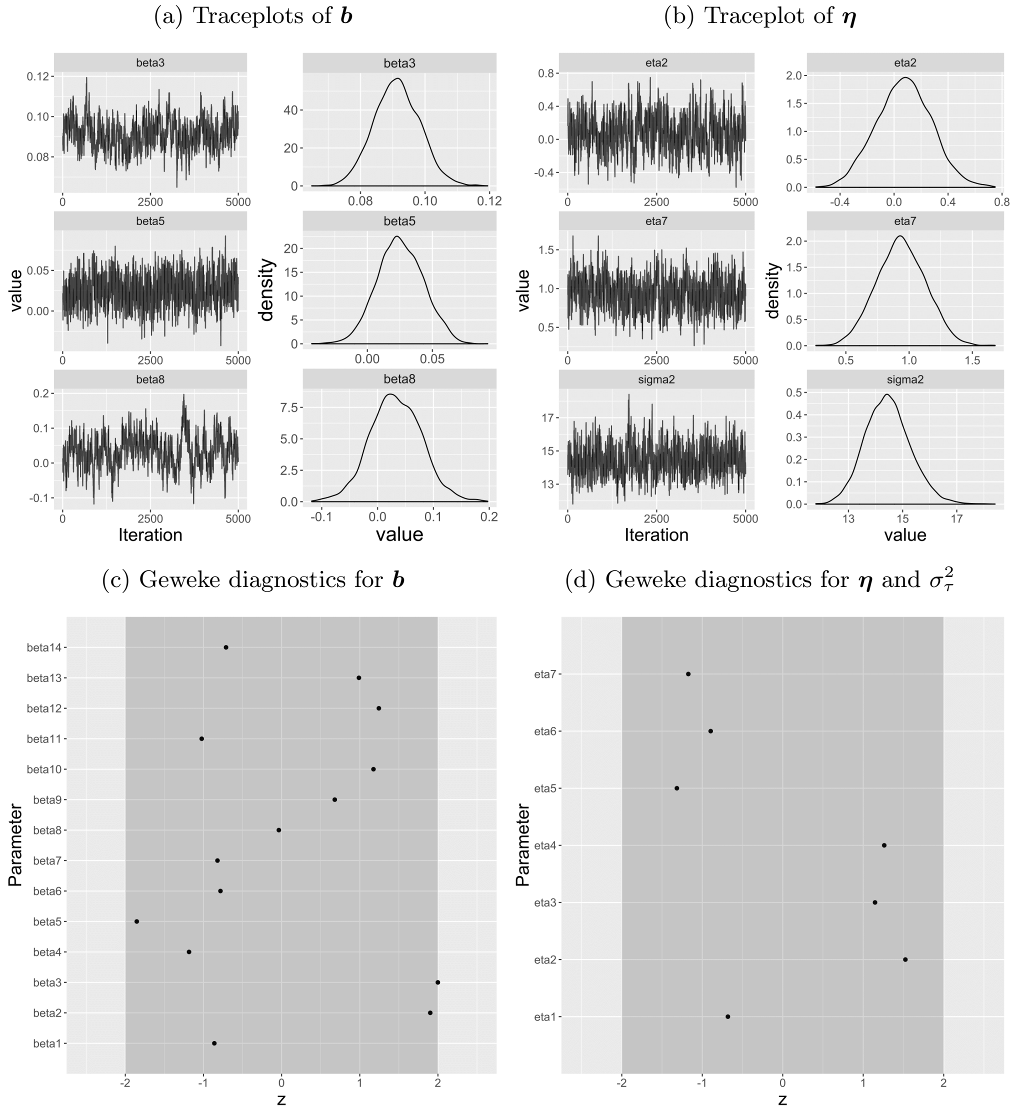
(b) Indegree distribution

(c) Receiver size distribution

(d) PPplot for time increments

Figure 7: Comparison of PPC results between log-normal (*red*) and exponential (*green*) distributions. Blue lines denote the observed statistics in (a)–(c) and denotes the diagonal line in (d).

Appendix D: Convergence diagnostics


(a) Traceplots of $\boldsymbol{b}$

(b) Traceplot of $\boldsymbol{\eta}$

(c) Geweke diagnostics for $\boldsymbol{b}$

(d) Geweke diagnostics for $\boldsymbol{\eta}$ and σ_τ^2

Figure 8: Convergence diagnostics from log-normal distribution.

References

Alemán, E. and Calvo, E. (2013). "Explaining policy ties in presidential congresses: A network analysis of bill initiation data." *Political Studies*, 61(2): 356–377. 2

ben Aaron, J., Denny, M., Desmarais, B., and Wallach, H. (2017). "Transparency by Conformity: A Field Experiment Evaluating Openness in Local Governments." *Public Administration Review*, 77(1): 68–77. 9

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). "Latent Dirichlet Allocation." *J. Mach. Learn. Res.*, 3: 993–1022. 1, 3

Bratton, K. A. and Rouse, S. M. (2011). "Networks in the legislative arena: How group dynamics affect cosponsorship." *Legislative Studies Quarterly*, 36(3): 423–460. 2

Burgess, A., Jackson, T., and Edwards, J. (2004). "Email overload: Tolerance levels of employees within the workplace." In *Innovations Through Information Technology: 2004 Information Resources Management Association International Conference, New Orleans, Louisiana, USA, May 23-26, 2004*, volume 1, 205. IGI Global. 1

Butts, C. T. (2008). "A RELATIONAL EVENT FRAMEWORK FOR SOCIAL ACTION." *Sociological Methodology*, 38(1): 155–200. 2

Camber Warren, T. (2010). "The geometry of security: Modeling interstate alliances as evolving networks." *Journal of Peace Research*, 47(6): 697–709. 2

Chatterjee, S., Diaconis, P., et al. (2013). "Estimating and understanding exponential random graph models." *The Annals of Statistics*, 41(5): 2428–2461. 1

Cranmer, S. J., Desmarais, B. A., and Kirkland, J. H. (2012a). "Toward a network theory of alliance formation." *International Interactions*, 38(3): 295–324. 2

Cranmer, S. J., Desmarais, B. A., and Menninga, E. J. (2012b). "Complex dependencies in the alliance network." *Conflict Management and Peace Science*, 29(3): 279–313. 2

Dai, B., Ding, S., Wahba, G., et al. (2013). "Multivariate bernoulli distribution." *Bernoulli*, 19(4): 1465–1483. 5

Desmarais, B. A. and Cranmer, S. J. (2017). "Statistical Inference in Political Networks Research." In Victor, J. N., Montgomery, A. H., and Lubell, M. (eds.), *The Oxford Handbook of Political Networks*. Oxford University Press. 2

Fahmy, C. and Young, J. T. (2016). "Gender Inequality and Knowledge Production in Criminology and Criminal Justice." *Journal of Criminal Justice Education*, 1–21. 2

Fellows, I. and Handcock, M. (2017). "Removing Phase Transitions from Gibbs Measures." In *Artificial Intelligence and Statistics*, 289–297. 5

Geweke, J. (2004). "Getting it right: Joint distribution tests of posterior simulators." *Journal of the American Statistical Association*, 99(467): 799–804. 7

Geweke, J. et al. (1991). *Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments*, volume 196. Federal Reserve Bank of Minneapolis, Research Department Minneapolis, MN, USA. 17

Hunter, D. R., Handcock, M. S., Butts, C. T., Goodreau, S. M., and Morris, M. (2008). "ergm: A package to fit, simulate and diagnose exponential-family models for networks." *Journal of statistical software*, 24(3): nihpa54860. 1

Hyndman, R. J. and Koehler, A. B. (2006). "Another look at measures of forecast accuracy." *International journal of forecasting*, 22(4): 679–688. 14

Kanungo, S. and Jain, V. (2008). "Modeling email use: a case of email system transition." *System Dynamics Review*, 24(3): 299–319. 1

Kinne, B. J. (2016). "Agreeing to arm: Bilateral weapons agreements and the global arms trade." *Journal of Peace Research*, 53(3): 359–377. 2

Krafft, P., Moore, J., Desmarais, B., and Wallach, H. M. (2012). "Topic-Partitioned Multinetwork Embeddings." In Pereira, F., Burges, C., Bottou, L., and Weinberger, K. (eds.), *Advances in Neural Information Processing Systems 25*, 2807–2815. Curran Associates, Inc. 1

Kronegger, L., Mali, F., Ferligoj, A., and Doreian, P. (2011). "Collaboration structures in Slovenian scientific communities." *Scientometrics*, 90(2): 631–647. 2

Lai, C.-H., She, B., and Tao, C.-C. (2017). "Connecting the dots: A longitudinal observation of relief organizations' representational networks on social media." *Computers in Human Behavior*, 74: 224–234. 2

Liang, X. (2015). "The Changing Impact of Geographic Distance: A Preliminary Analysis on the Co-author Networks in Scientometrics (1983-2013)." In *System Sciences (HICSS), 2015 48th Hawaii International Conference on*, 722–731. IEEE. 2

Lim, K. W., Chen, C., and Buntine, W. (2013). "Twitter-Network Topic Model: A Full Bayesian Treatment for Social Network and Text Modeling." In *NIPS2013 Topic Model workshop*, 1–5. 1

McCallum, A., Corrada-Emmanuel, A., and Wang, X. (2005). "The Author-Recipient-Topic Model for Topic and Role Discovery in Social Networks, with Application to Enron and Academic Email." In *Workshop on Link Analysis, Counterterrorism and Security*, 33. 1

Neal, P. and Kypraios, T. (2015). "Exact Bayesian inference via data augmentation." *Statistics and Computing*, 25(2): 333–347. 7

Nelder, J. A. and Baker, R. J. (1972). *Generalized linear models*. Wiley Online Library. 5

Peng, T.-Q., Liu, M., Wu, Y., and Liu, S. (2016). "Follower-followee network, communication networks, and vote agreement of the US members of congress." *Communication Research*, 43(7): 996–1024. 2

Perry, P. O. and Wolfe, P. J. (2013). "Point process modelling for directed interaction networks." *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 75(5): 821–849. 2, 10

Pew, R. C. (2016). "Social Media Fact Sheet." *Accessed on 03/07/17*. 1

Rao, P. (2000). "Applied survival analysis: regression modeling of time to event data." *Journal of the American Statistical Association*, 95(450): 681–681. 6

Rizopoulos, D. (2012). *Joint models for longitudinal and time-to-event data: With applications in R*. CRC Press. 6

Robins, G., Pattison, P., Kalish, Y., and Lusher, D. (2007). "An introduction to exponential random graph (p*) models for social networks." *Social networks*, 29(2): 173–191. 1

Rubin, D. B. et al. (1984). "Bayesianly justifiable and relevant frequency calculations for the applied statistician." *The Annals of Statistics*, 12(4): 1151–1172. 15

Schein, A., Zhou, M., Blei, D. M., and Wallach, H. (2016). "Bayesian poisson tucker decomposition for learning the structure of international relations." *arXiv preprint arXiv:1606.01855*. 3

Snijders, T., Steglich, C., and Schweinberger, M. (2007). *Modeling the coevolution of networks and behavior*. na. 12

Snijders, T. A. (1996). "Stochastic actor-oriented models for network change." *Journal of mathematical sociology*, 21(1-2): 149–172. 2, 6, 12

Szóstek, A. M. (2011). "?Dealing with My Emails?: Latent user needs in email management." *Computers in Human Behavior*, 27(2): 723–729. 1

Tanner, M. A. and Wong, W. H. (1987). "The calculation of posterior distributions by data augmentation." *Journal of the American statistical Association*, 82(398): 528–540. 7

Yoon, H. Y. and Park, H. W. (2014). "Strategies affecting Twitter-based networking pattern of South Korean politicians: social network analysis and exponential random graph model." *Quality & Quantity*, 1–15. 2