arXiv:1507.02293v1 [cs.SI] 8 Jul 2015

COEVOLVE: A Joint Point Process Model for Information Diffusion and Network Co-evolution

Mehrdad Farajtabar[1], Yichen Wang[1], Manuel Gomez Rodriguez[2], Shuang Li[1], Hongyuan Zha[1], and Le Song[1]

[1]Georgia Institute of Technology, {mehrdad,yichen.wang,sli370}@gatech.edu, {zha,lsong}@cc.gatech.edu
[2]Max Plank Institute for Software Systems, manuelgr@mpi-sws.org

Abstract

Information diffusion in online social networks is affected by the underlying network topology, but it also has the power to change it. Online users are constantly creating new links when exposed to new information sources, and in turn these links are alternating the way information spreads. However, these two highly intertwined stochastic processes, information diffusion and network evolution, have been predominantly studied *separately*, ignoring their co-evolutionary dynamics.

We propose a temporal point process model, COEVOLVE, for such joint dynamics, allowing the intensity of one process to be modulated by that of the other. This model allows us to efficiently simulate interleaved diffusion and network events, and generate traces obeying common diffusion and network patterns observed in real-world networks. Furthermore, we also develop a convex optimization framework to learn the parameters of the model from historical diffusion and network evolution traces. We experimented with both synthetic data and data gathered from Twitter, and show that our model provides a good fit to the data as well as more accurate predictions than alternatives.

1 Introduction

Online social networks, such as Twitter or Weibo, have become large information networks where people share, discuss and search for information of personal interest as well as breaking news [1]. In this context, users often forward to their *followers* information they are exposed to via their *followees*, triggering the emergence of information *cascades* that travel through the network [2], and constantly create new links to information sources, triggering changes in the network itself over time. Importantly, recent empirical studies with Twitter data have shown that both information diffusion and network evolution are coupled and network changes are often triggered by information diffusion [3, 4, 5].

While there have been many recent works on modeling information diffusion [2, 6, 7, 8, 9] and network evolution [10, 11, 12], most of them treat these two stochastic processes independently and separately, ignoring the influence one may have on the other over time. Thus, to better understand information diffusion and network evolution, there is an urgent need for joint probabilistic models of the two processes, which are largely inexistent to date.

In this paper, we propose a probabilistic generative model, COEVOLVE, for the joint dynamics of information diffusion and network evolution. Our model is based on the framework of temporal point processes, which explicitly characterize the continuous time interval between events, and it consists of two interwoven and interdependent components, as shown in Figure 1:

I. **Information diffusion process.** We design an "identity revealing" multivariate Hawkes process [13] to capture the mutual excitation behavior of retweeting events, where the intensity of such events in a user is boosted by previous events from her time-varying set of followees. Although Hawkes processes have been used for information diffusion before [14, 15, 16, 17, 18, 19, 20, 21], the key innovation of our approach is to explicitly model the excitation due to a particular source node, hence revealing the identity of the source. Such

design reflects the reality that information sources are explicitly acknowledged, and it also allows a particular information source to acquire new links in a rate according to her "informativeness".

II. **Network evolution process.** We model link creation as an "information driven" survival process, and couple the intensity of this process with retweeting events. Although survival processes have been used for link creation before [22, 23], the key innovation in our model is to incorporate retweeting events as the driving force for such processes. Since our model has captured the source identity of each retweeting event, new links will be targeted toward the information sources, with an intensity proportional to their degree of excitation and each source's influence.

Our model is designed in such a way that it allows the two processes, information diffusion and network evolution, unfold simultaneously in the same time scale and excise bidirectional influence on each other, allowing sophisticated coevolutionary dynamics to be generated (*e.g.*, see Figures 1 and 6). Importantly, the flexibility of our model does not prevent us from efficiently simulating diffusion and link events from the model and learning its parameters from real world data:

- **Efficient simulation.** We design a scalable sampling procedure that exploits the sparsity of the generated networks. Its complexity is $O(nd \log m)$, where n is the number of samples, m is the number of nodes and d is the maximum number of followees per user.
- **Convex parameters learning.** We show that the model parameters that maximize the joint likelihood of observed diffusion and link creation events can be found via convex optimization.

Finally, we experimentally verify that our model can produce coevolutionary dynamics of information diffusion and network evolution, and generate retweet and link events that obey common information diffusion patterns (*e.g.*, cascade structure, size and depth), static network patterns (*e.g.*, node degree) and temporal network patterns (*e.g.*, shrinking diameter) described in related literature [24, 11, 25]. Furthermore, we show that, by modeling the coevolutionary dynamics, our model provide significantly more accurate link and diffusion event predictions than alternatives in large scale Twitter dataset [3].

1.1 Further Related Work

The works most closely related to ours are empirical studies of information diffusion and network evolution [26, 27, 5, 3, 4]. Among them, [5] was the first to show experimental evidence the likelihood that a user u starts following a user s increases with the number of messages from s seen by u. [3] investigated the temporal and statistical characteristics of retweet-driven connections within the Twitter network and then identified the number of retweets as a key factor to infer such connections. Authors in [4] showed that the Twitter network can be characterized by steady rates of change, interrupted by sudden bursts of new connections, triggered by retweet cascades. They also developed a method to predict which retweets are more likely to trigger these bursts. Finally, [28] utilized multivariate Hawkes process to establish a connection between temporal properties of activities and the structure of the network. In contrast to our work they studied the static properties, *e.g.*, community structure and inferred the latent clusters using the observed activities.

However, there are fundamental differences between the above mentioned studies and our work. First, they only characterize the effect that information diffusion has on the network dynamics, but not the bidirectional influence. In contrast, our probabilistic generative model takes into account the bidirectional influence between information diffusion and network dynamics. Second, previous studies are mostly empirical and only make binary predictions on link creation events. For example, [5, 3] predict whether a new link will be created based on the number of retweets; [4] predict whether a burst of new links will occur based on the number of retweets and users' similarity. In contrast, our model can learn parameters from real world data, and predict the precise timing of both diffusion and new link events.

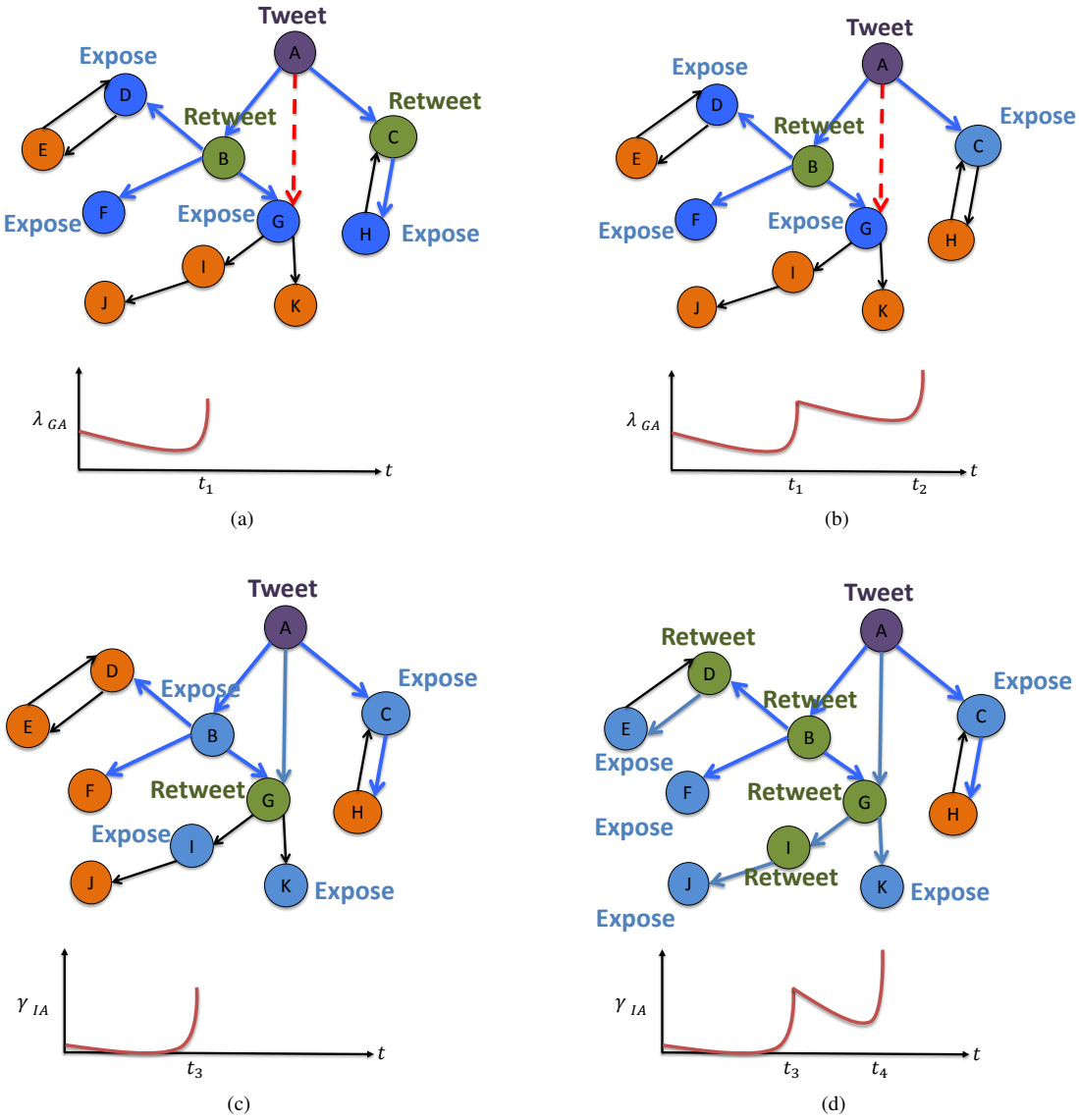


Figure 1: Joint dynamics of information diffusion and network evolution. Node A is the source of information and node G is a potential follower. Blue links form the diffusion paths. Green and blue circles are nodes which see the information. Green circles are those who re-share and propagate the information. Orange circles are nodes unaware of A's post. In Panel (a), node G is first exposed to a piece of information originated from A via a retweet from B at time t_1. The intensity $\lambda^*_{GA}(t)$ of G establishing a link to A increases. In Panel (b), B retweets another tweet posted by A at time t_2, node G is exposed again to information from A and the intensity $\lambda^*_{GA}(t)$ increases again. After G observes sufficient information from A, she finds her a valuable source and thus G decides to follow A directly. This is how information diffusion affects network formation. On the other side, network evolution will affect information diffusion, as shown in Panels (c-d). After G decides to follow A directly, she opens a new path of information for her downstream nodes. Now, when G retweets tweets originally posted by A, node I will be exposed and may retweet them too. Here, the exposure at time t_3 and t_4 is sufficient for I to retweet a tweet originated by A.

2 Backgrounds on Temporal Point Processes

A temporal point process is a random process whose realization consists of a list of discrete events localized in time, $\{t_i\}$ with $t_i \in \mathbb{R}^+$ and $i \in \mathbb{Z}^+$. Many different types of data produced in online social networks can be represented as temporal point processes, such as the times of retweets and link creations. A temporal point process can be equivalently represented as a counting process, $N(t)$, which records the number of events before time t. Let the history $\mathcal{H}(t)$ be the list of times of events $\{t_1, t_2, \ldots, t_n\}$ up to but not including time t. Then, the number of observed events in a small time window dt between $[t, t + dt)$ is $dN(t) = \sum_{t_i \in \mathcal{H}(t)} \delta(t - t_i)\, dt$, and hence $N(t) = \int_0^t dN(s)$, where $\delta(t)$ is a Dirac delta function. More generally, given a function $f(t)$, we can define the convolution with respect to $dN(t)$ as

$$f(t) \star dN(t) := \int_0^t f(t - \tau)\, dN(\tau) = \sum\nolimits_{t_i \in \mathcal{H}(t)} f(t - t_i). \tag{1}$$

The point process representation of temporal data is fundamentally different from the discrete time representation typically used in social network analysis. It directly models the time interval between events as random variables, and avoid the need to pick a time window to aggregate events. It allows temporal events to be modeled in a more fine grained fashion, and has a remarkably rich theoretical support [29].

An important way to characterize temporal point processes is via the conditional intensity function — a stochastic model for the time of the next event given all the times of previous events. Formally, the conditional intensity function $\lambda^*(t)$ (intensity, for short) is the conditional probability of observing an event in a small window $[t, t + dt)$ given the history $\mathcal{H}(t)$, *i.e.*,

$$\lambda^*(t)dt := \mathbb{P}\left\{\text{event in } [t, t + dt) | \mathcal{H}(t)\right\} = \mathbb{E}[dN(t) | \mathcal{H}(t)], \tag{2}$$

where one typically assumes that only one event can happen in a small window of size dt, *i.e.*, $dN(t) \in \{0, 1\}$. Then, given a time $t' \geqslant t$, we can also characterize the conditional probability that no event happens during $[t, t')$ and the conditional density that an event occurs at time t' as $S^*(t') = \exp(-\int_t^{t'} \lambda^*(\tau)\, d\tau)$ and $f^*(t') = \lambda^*(t')\, S^*(t')$ respectively [29]. Furthermore, we can express the log-likelihood of a list of events $\{t_1, t_2, \ldots, t_n\}$ in an observation window $[0, T)$ as

$$\mathfrak{L} = \sum_{i=1}^{n} \log \lambda^*(t_i) - \int_0^T \lambda^*(\tau)\, d\tau, \quad T \geqslant t_n. \tag{3}$$

This simple log-likelihood will later enable us to learn the parameters of our model from observed data.

Finally, the functional form of the intensity $\lambda^*(t)$ is often designed to capture the phenomena of interests. Some useful functional forms we will use later are [29]:

(i) **Poisson process.** The intensity is assumed to be independent of the history $\mathcal{H}(t)$, but it can be a time-varying function, *i.e.*, $\lambda^*(t) = g(t) \geqslant 0$;

(ii) **Hawkes Process.** The intensity models a mutual excitation between events, *i.e.*,

$$\lambda^*(t) = \mu + \alpha \kappa_\omega(t) \star dN(t) = \mu + \alpha \sum\nolimits_{t_i \in \mathcal{H}(t)} \kappa_\omega(t - t_i), \tag{4}$$

where $\kappa_\omega(t) := \exp(-\omega t)\mathbb{I}[t \geqslant 0]$ is an exponential triggering kernel, $\mu \geqslant 0$ is a baseline intensity independent of the history. Here, the occurrence of each historical event increases the intensity by a certain amount determined by the kernel and the weight $\alpha \geqslant 0$, making the intensity history dependent and a stochastic process by itself. We will focus on the exponential kernel in this paper. However, other functional forms for the triggering kernel, such as log-logistic function, are possible, and our model does not depend on this particular choice; and,

(iii) **Survival process.** There is only one event for an instantiation of the process, *i.e.*,

$$\lambda^*(t) = g^*(t)(1 - N(t)), \tag{5}$$

where $\lambda^*(t)$ becomes 0 if an event already happened before t.

3 Generative Model of Information Diffusion and Network Co-evolution

In this section, we use the above background on temporal point processes to formulate our probabilistic generative model for the joint dynamics of information diffusion and network evolution.

3.1 Event Representation

We model the generation of two types of events: tweet/retweet events, e^r, and link creation events, e^l. Instead of just the time t, we record each event as a triplet

$$e^r \text{ or } e^l := (\underset{\uparrow}{u}, \overset{\text{source}}{\underset{\uparrow}{s}}, \underset{\uparrow}{t}). \quad (6)$$
$$\text{destination} \quad \text{time}$$

For retweet event, the triplet means that the destination node u retweets at time t a tweet originally posted by source node s. Recording the source node s reflects the real world scenario that information sources are explicitly acknowledged. Note that the occurrence of event e^r does *not* mean that u is directly retweeting from or is connected to s. This event can happen when u is retweeting a message by another node u' where the original information source s is acknowledged. Node u will pass on the same source acknowledgement to its followers (*e.g.*, "I agree @a @b @c @s"). Original tweets posted by node u are allowed in this notation. In this case, the event will simply be $e^r = (u, u, t)$. Given a list of retweet events up to but not including time t, the history $\mathcal{H}^r_{us}(t)$ of retweets by u due to source s is $\mathcal{H}^r_{us}(t) = \{e^r_i = (u_i, s_i, t_i) | u_i = u \text{ and } s_i = s\}$. The entire history of retweet events is denoted as $\mathcal{H}^r(t) := \cup_{u,s\in[m]}\mathcal{H}^r_{us}(t)$.

For link creation event, the triplet means that destination node u creates at time t a link to source node s, *i.e.*, from time t on, node u starts following node s. To ease the exposition, we restrict ourselves to the case where links cannot be deleted and thus each (directed) link is created only once. However, our model can be easily augmented to consider multiple link creations and deletions per node pair, as discussed in Section 9. We denote the link creation history as $\mathcal{H}^l(t)$.

3.2 Joint Model with Two Interwoven Components

Given m users, we use two sets of counting processes to record the generated events, one for information diffusion and the other for network evolution. More specifically,

I. Retweet events are recorded using a matrix $\boldsymbol{N}(t)$ of size $m \times m$ for each fixed time point t. The (u, s)-th entry in the matrix, $N_{us}(t) \in \{0\} \cup \mathbb{Z}^+$, counts the number of retweets of u due to source s up to time t. These counting processes are "identity revealing", since they keep track of the source node that triggers each retweet. This matrix $\boldsymbol{N}(t)$ can be dense, since $N_{us}(t)$ can be nonzero even when node u does not directly follow s. We also let $d\boldsymbol{N}(t) := (\, dN_{us}(t)\,)_{u,s\in[m]}$.

II. Link events are recorded using an adjacency matrix $\boldsymbol{A}(t)$ of size $m \times m$ for each fixed time point t. The (u, s)-th entry in the matrix, $A_{us}(t) \in \{0, 1\}$, indicates whether u is directly following s. That is $A_{us}(t) = 1$ means the directed link has been created before t. For simplicity of exposition, we do not allow self-links. The matrix $\boldsymbol{A}(t)$ is typically sparse, but the number of nonzero entries can change over time. We also define $d\boldsymbol{A}(t) := (\, dA_{us}(t)\,)_{u,s\in[m]}$.

Then the interwoven information diffusion and network evolution processes can be characterized using their respective intensities $\mathbb{E}[d\boldsymbol{N}(t) \,|\, \mathcal{H}^r(t) \cup \mathcal{H}^l(t)] = \boldsymbol{\Gamma}^*(t)\, dt$ and $\mathbb{E}[d\boldsymbol{A}(t) \,|\, \mathcal{H}^r(t) \cup \mathcal{H}^l(t)] = \boldsymbol{\Lambda}^*(t)\, dt$, where $\boldsymbol{\Gamma}^*(t) = (\, \gamma^*_{us}(t)\,)_{u,s\in[m]}$ and $\boldsymbol{\Lambda}^*(t) = (\, \lambda^*_{us}(t)\,)_{u,s\in[m]}$. The sign * means that the intensity matrices will depend on the joint history, $\mathcal{H}^r(t) \cup \mathcal{H}^l(t)$, and hence their evolution will be coupled. By this coupling, we make: (i) the counting processes for link creation to be "information driven" and (ii) the evolution of the linking structure to change the information diffusion process. Figure 1 illustrates our joint model. In the next two sections, we will specify the details of these two intensity matrices.

3.3 Information Diffusion Process

We model the intensity, $\mathbf{\Gamma}^*(t)$, for retweeting events using multivariate Hawkes process [13]:

$$\gamma^*_{us}(t) = \mathbb{I}[u = s]\, \eta_u + \mathbb{I}[u \neq s]\, \beta_s \sum\nolimits_{v \in \mathcal{F}_u(t)} \kappa_{\omega_1}(t) \star (A_{uv}(t)\ dN_{vs}(t))\,, \tag{7}$$

where $\mathbb{I}[\cdot]$ is the indicator function and $\mathcal{F}_u(t) := \{v \in [m] : A_{uv}(t) = 1\}$ is the current set of followees of u. The term $\eta_u \geqslant 0$ is the intensity of original tweets by a user u on his own initiative, becoming the source of a cascade and the term $\beta_s \sum_{v \in \mathcal{F}_u(t)} \kappa_\omega(t) \star (A_{uv}(t)\ dN_{vs}(t))$ models the propagation of peer influence over the network, where the triggering kernel $\kappa_{\omega_1}(t)$ models the decay of peer influence over time.

Note that the retweet intensity matrix $\mathbf{\Gamma}^*(t)$ is by itself a stochastic process that depends on the time-varying network topology, the non-zero entries in $\boldsymbol{A}(t)$, whose growth is controlled by the network evolution process in Section 3.4. Hence the model design captures the influence of the network topology and each source's influence, β_s, on the information diffusion process. More specifically, to compute $\gamma^*_{us}(t)$, one first finds the current set $\mathcal{F}_u(t)$ of followees of u, and then aggregates the retweets of these followees that are due to source s. Note that these followees may or may not *directly* follow source s. Then, the more frequently node u is exposed to retweets of tweets originated from source s via her followees, the more likely she will also retweet a tweet originated from source s. Once node u retweets due to source s, the corresponding $N_{us}(t)$ will be incremented, and this in turn will increase the likelihood of triggering retweets due to source s among the followers of u. Thus, the source does *not* simply broadcast the message to nodes directly following her but her influence propagates through the network even to those nodes that do not directly follow her. Finally, this information diffusion model allows a node to repeatedly generate events in a cascade, and is very different from the independent cascade or linear threshold models [30] which allow at most one event per node per cascade.

3.4 Network Evolution Process

We model the intensity, $\mathbf{\Lambda}^*(t)$, for link creation using a combination of survival and Hawkes process:

$$\lambda^*_{us}(t) = (1 - A_{us}(t))(\mu_u + \alpha_u\, \kappa_{\omega_2}(t) \star dN_{us}(t)) \tag{8}$$

where the term $1 - A_{us}(t)$ effectively ensures a link is created only once, and after that, the corresponding intensity is set to zero. The term $\mu_u \geqslant 0$ denotes a baseline intensity, which models when a node u decides to follow a source s spontaneously at her own initiative. The term $\alpha_u \kappa_{\omega_2}(t) \star dN_{us}(t)$ corresponds to the retweets of node u due to tweets originally published by source s, where the triggering kernel $\kappa_{\omega_2}(t)$ models the decay of interests over time. Here, the higher the corresponding retweet intensity, the more likely u will find information by source s useful and will create a *direct* link to s.

The link creation intensity $\mathbf{\Lambda}^*(t)$ is also a stochastic process by itself, which depends on the retweet events, and is driven by the retweet count increments $dN_{us}(t)$. It captures the influence of retweets on the link creation, and closes the loop of mutual influence between information diffusion and network topology.

Note that creating a link is more than just adding a path or allowing information sources to take shortcuts during diffusion. The network evolution makes fundamental changes to the diffusion dynamics and stationary distribution of the diffusion process in Section 3.3. As shown in [15], given a fixed network structure $\boldsymbol{A}$, the expected retweet intensity $\boldsymbol{\mu}_s(t)$ at time t due to source s will depend of the network structure in a highly nonlinear fashion, *i.e.*, $\boldsymbol{\mu}_s(t) := \mathbb{E}[\mathbf{\Gamma}^*_{:s}(t)] = (e^{(\boldsymbol{A} - \omega_1 \boldsymbol{I})t} + \omega_1 (\boldsymbol{A} - \omega_1 \boldsymbol{I})^{-1} (e^{(\boldsymbol{A} - \omega_1 \boldsymbol{I})t} - \boldsymbol{I}))\, \boldsymbol{\eta}_s$, where $\boldsymbol{\eta}_s \in \mathbb{R}^m$ has a single nonzero entry with value η_s and $e^{(\boldsymbol{A} - \omega_1 \boldsymbol{I})t}$ is the matrix exponential. When $t \to \infty$, the stationary intensity $\bar{\boldsymbol{\mu}}_s = (\boldsymbol{I} - \boldsymbol{A}/\omega)^{-1}\, \boldsymbol{\eta}_s$ is also nonlinearly related to the network structure. Thus given two network structures $\boldsymbol{A}(t)$ and $\boldsymbol{A}(t')$ at two points in time, which are different by a few edges, the effect of these edges on the information diffusion is not just simply an additive relation. Depending on how these newly created edges modify the eigen-structure of the sparse matrix $\boldsymbol{A}(t)$, their effect can be drastic to the information diffusion.

Algorithm 1 Efficient Simulation

Initialization:
Initialize the priority queue Q
for $\forall\, u, s \in [m]$ **do**
  Sample next link event e^l_{us} from A_{us} (Algorithm 3)
  $Q.insert(e^l_{us})$
  Sample next retweet event e^r_{us} from N_{us} (Algorithm 3)
  $Q.insert(e^r_{us})$
General Subroutine:
$t \leftarrow 0$
while $t < T$ **do**
  $e \leftarrow Q.extract\_min()$
  if $e = (u, s, t')$ is a retweet event **then**
    Update the history $\mathcal{H}^r_{us}(t') = \mathcal{H}^r_{us}(t) \cup \{e\}$
    for $\forall\, v\ s.t.\ u \rightsquigarrow v$ **do**
      Update event intensity: $\gamma_{vs}(t') = \gamma_{vs}(t'^{-}) + \beta$
      Sample retweet event e^r_{vs} from N_{vs} (Algorithm 3)
      $Q.update\_key(e^r_{vs})$
      if NOT $s \rightsquigarrow v$ **then**
        Update link intensity: $\lambda_{vs}(t') = \lambda_{vs}(t'^{-}) + \alpha$
        Sample link event e^l_{vs} from A_{vs} (Algorithm 3)
        $Q.update\_key(e^l_{vs})$
  else
    Update the history $\mathcal{H}^l_{us}(t') = \mathcal{H}^l_{us}(t) \cup \{e\}$
    $\lambda_{us}(t) \leftarrow 0\ \ \forall\, t > t'$
  $t \leftarrow t'$

4 Efficient Simulation of Coevolutionary Dynamics

We can simulate samples (link creations, tweets and retweets) from our model by adapting Ogata's thinning algorithm [31], originally designed for multidimensional Hawkes processes. However, a naive implementation of Ogata's algorithm would scale poorly, *i.e.*, for each sample, we would need to re-evaluate $\mathbf{\Gamma}^*(t)$ and $\mathbf{\Lambda}^*(t)$, thus, to draw n samples, we would need to perform $O(m^2n^2)$ operations, where m is the number of nodes.

We designed a sampling procedure that is especially well-fitted for the structure of our model. The algorithm is based on the following key idea: if we consider each intensity function in $\mathbf{\Gamma}^*(t)$ and $\mathbf{\Lambda}^*(t)$ as a separate Hawkes process and draw a sample from each, it is easy to show that the minimum among all these samples is a valid sample from the model [13]. However, by drawing samples from all intensities, the computational complexity would not improve. However, when the network is sparse, whenever we sample a new node (or link) event from the model, only a small number of intensity functions, in the local neighborhood of the node (or the link), will change. As a consequence, we can reuse most of the samples from the intensity functions for the next new sample and find which intensity functions we need to change in $O(\log m)$ operations, using a heap. Finally, we exploit the properties of the exponential function to update individual intensities for each new sample in $O(1)$: let t_i and t_{i+1} be two consecutive events, then, we can compute $\lambda^*(t_{i+1})$ as $(\lambda^*(t_i) - \mu)\exp(-\omega(t_{i+1} - t_i)) + \mu$ without the need to compare all previous events.

The complete simulation algorithm is summarized in Algorithm 1. By using Algorithm 1, we reduce the complexity from $O(n^2m^2)$ to $O(nd\log m)$, where d is the maximum number of followees per node. That means, our algorithm scales logarithmically with the number of nodes and linearly with the number of edges at any point in time during the simulation. We also note that the events for link creations, tweets and retweets are generated in a temporally intertwined and interleaving fashion by Algorithm 1. This is because every new retweet event will modify the intensity for link creation, and after each link creation we also need to update the retweet intensities.

Algorithm 2 Efficient Intensity Computation

Global Variabels:
Last time of intensity computation: t
Last value of intensity computation: I
Initialization:
$t = 0, I = \mu$
function $get\_intensity(t')$
  $I' = (I - \mu)\exp(-\omega(t' - t)) + \mu$
  $t = t', I = I'$
  return I
end function

Algorithm 3 Sampling

Input: Current time: t
Output: Next event time: s
Set $s \leftarrow t$ and $\hat{\lambda} \leftarrow \lambda(s)$ (Algorithm 2)
while $s < T$ **do**
  Generate $g \sim exp(\hat{\lambda})$ set $s \leftarrow s + g$
  Set $\bar{\lambda} \leftarrow \lambda(s)$ (Algorithm 2)
  Rejection test: Generate $d \sim \mathcal{U}(0, 1)$
  if $d \times \hat{\lambda} < \bar{\lambda}$ **then** **return** s **else** $\hat{\lambda} = \bar{\lambda}$
return s

5 Efficient Parameter Estimation from Coevolutionary Events

Given a collection of retweet events $\mathcal{E} = \{e_i^r\}$ and link creation events $\mathcal{A} = \{e_i^l\}$ recorded within a time window $[0, T)$, we can easily estimate the parameters needed in our model using maximum likelihood estimation. Here, we compute the joint log-likelihood $\mathfrak{L}(\{\mu_u\}, \{\alpha_u\}, \{\eta_u\}, \{\beta_s\})$ of these events using Eq. (3), *i.e.*,

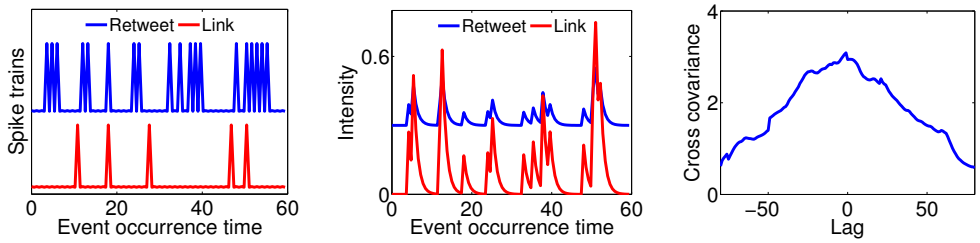
$$\underbrace{\sum_{e_i^r \in \mathcal{E}} \log\left(\gamma^*_{u_i s_i}(t_i)\right) - \sum_{u,s \in [m]} \int_0^T \gamma^*_{us}(\tau)\, d\tau}_{\text{tweet / retweet}} + \underbrace{\sum_{e_i^l \in \mathcal{A}} \log\left(\lambda^*_{u_i s_i}(t_i)\right) - \sum_{u,s \in [m]} \int_0^T \lambda^*_{us}(\tau)\, d\tau}_{\text{links}}. \tag{9}$$

For the terms corresponding to retweets, the log term only sums over the actual observed events, but the integral term actually sums over all possible combination of destination and source pairs, even if there is no event between a particular pair of destination and source. For such pairs with no observed events, the corresponding counting processes have essentially survived the observation window $[0, T)$, and the term $-\int_0^T \gamma^*_{us}(\tau)d\tau$ simply corresponds to the log survival probability. Terms corresponding to links have a similar structure to those for retweet.

Since $\gamma^*_{us}(t)$ and λ^*_{us} are linear in the parameters (η_u, β_s) and (μ_u, α_u) respectively, then $\log(\gamma^*_{us}(t))$ and $\log(\lambda^*_{us})$ are concave functions in these parameters. Integration of $\gamma^*_{us}(t)$ and λ^*_{us} still results in linear functions of the parameters. Thus the overall objective in Eq. (9) is concave, and the global optimum can be found by many algorithms. In our experiments, we adapt the efficient algorithm developed in previous work [20, 21]. Furthermore, the optimization problem decomposes in m independent problems, one per node u, and can be readily parallelized.

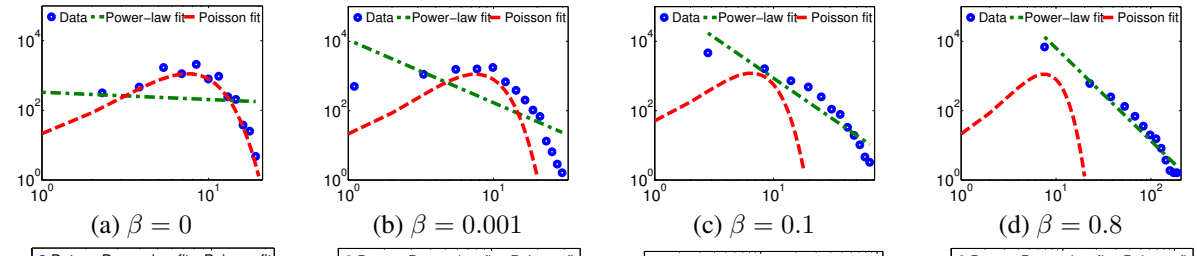
6 Properties of Simulated Co-evolution, Networks and Cascades

In this section, we perform an empirical investigation of the properties of the networks and information cascades generated by our model. In particular, we show that our model can generate co-evolutionary retweet and link dynamics and a wide spectrum of static and temporal network patterns and information cascades.



(a) (b) (c)

Figure 2: Coevolutionary dynamics for synthetic data. a) Spike trains of link and retweet events. b) Link and retweet intensities. c) Cross covariance of link and retweet intensities.



(a) $\beta = 0$ (b) $\beta = 0.001$ (c) $\beta = 0.1$ (d) $\beta = 0.8$

(a) $\alpha = 0$ (b) $\alpha = 0.05$ (c) $\alpha = 0.1$ (d) $\alpha = 0.2$

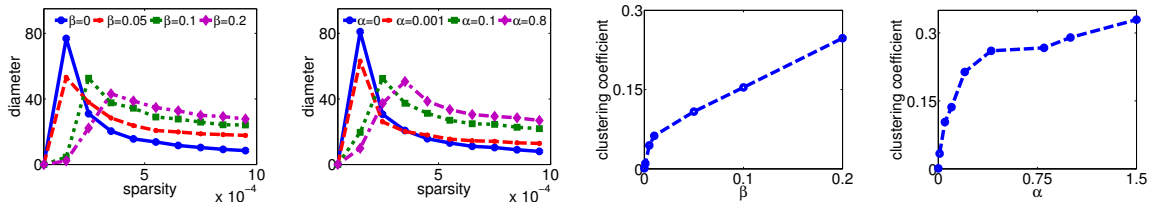
Figure 3: Degree distributions when network sparsity level reaches 0.001 for different β (α) values and fixed $\alpha = 0.1$ ($\beta = 0.1$).

6.1 Simulation Settings

Throughout this section, we simulate the evolution of a 8,000-node network as well as the propagation of information over the network by sampling from our model using Algorithm 1. We set the exogenous intensities of the link event and diffusion event intensities to $\mu_u = \mu = 4 \times 10^{-6}$ and $\eta_u = \eta = 1.5$ respectively, and the triggering kernel parameter to $\omega_1 = \omega_2 = 1$. The parameter μ determines the independent growth of the network, roughly speaking, the expected number of links each user establishes spontaneously before time T is μT. Whenever we investigate a static property, we choose the same sparsity level of 0.001.

6.2 Retweet and Link Coevolution

Figures 2(a,b) visualize the retweet and link events, aggregated across different sources, and the corresponding intensities for one node and one realization, picked at random. Here, it is already apparent that retweets and link creations are clustered in time and often follow each other. Further, Figure 2(c) shows the cross-covariance of the retweet and link creation intensity, computed across multiple realizations, for the same node, *i.e.*, if $f(t)$ and $g(t)$ are two intensities, the cross-covariance is a function of the time lag τ defined as $h(\tau) = \int f(t+\tau)g(t)\,dt$. It can be seen that the cross-covariance has its peak around 0, *i.e.*, retweets and link creations are highly correlated and co-evolve over time. For ease of exposition, we illustrated co-evolution using one node, however, we found consistent results across nodes.



(a) Diameter, $\alpha = 0.1$ (b) Diameter, $\beta = 0.1$ (c) CC, $\alpha = 0.1$ (d) CC, $\beta = 0.1$

Figure 4: Diameter and clustering coefficient for network sparsity 0.001. Panels (a) and (b) show the diameter against sparsity over time for fixed $\alpha = 0.1$, and for fixed $\beta = 0.1$ respectively. Panels (c) and (d) show the clustering coefficient (CC) against β and α, respectively.

6.3 Degree Distribution

Empirical studies have shown that the degree distribution of online social networks and microblogging sites follow a power law [10, 1], and argued that it is a consequence of the rich get richer phenomena. The degree distribution of a network is a power law if the expected number of nodes m_d with degree d is given by $m_d \propto d^{-\gamma}$, where $\gamma > 0$. Intuitively, the higher the values of the parameters α and β, the closer the resulting degree distribution follows a power-law; the lower their values, the closer the distribution to an Erdos-Renyi random graph [32]. Figure 3 confirms this intuition by showing the degree distribution for different values of β and α.

6.4 Small (shrinking) Diameter

There is empirical evidence that the diameter of online social networks and microblogging sites exhibit relatively small diameter and shrinks (or flattens) as the network grows [33, 10, 24]. Figures 4(a-b) show the diameter on the largest connected component (LCC) against the sparsity of the network over time for different values of α and β. Although at the beginning, there is a short increase in the diameter due to the merge of small connected components, the diameter decreases as the network evolves. Here, nodes *arrive* to the network when they follow (or are followed by) a node in the largest connected component.

6.5 Clustering Coefficient

Triadic closure [34, 12, 35] has been often presented as a plausible link creation mechanism. However, different social networks and microblogging sites present different levels of triadic closure [36]. Importantly, our method is able to generate networks with different levels of triadic closure, as shown by Figure 4(c-d), where we plot the clustering coefficient [37], which is proportional to the frequency of triadic closure, for different values of α and β.

6.6 Network Visualization

Figure 5 visualizes several snapshots of the largest connected component (LCC) of two 300-node networks for two particular realizations of our model, under two different values of β. In both cases, we used $\mu = 2 \times 10^{-4}$, $\alpha = 1$, and $\eta = 1.5$. The top two rows correspond to $\beta = 0$ and represent one end of the spectrum, *i.e.*, Erdos-Renyi random network. Here, the network evolves uniformly. The bottom two rows correspond to $\beta = 0.8$ and represent the other end, *i.e.*, scale-free networks. Here, the network evolves locally, and clusters emerge naturally as a consequence of the local growth. They are depicted using a combination of forced directed and Fruchterman Reingold layout with Gephi[1]. This figure clearly illustrates that by careful choice of parameters we can generate networks with a very different structure.

Furthermore, the retweet events (from others as source) of two nodes, A and B, are demonstrated at the bottom row. These two nodes arrive almost at the same time and establish links to two other nodes. However, node A's followees

[1] http://gephi.github.io/

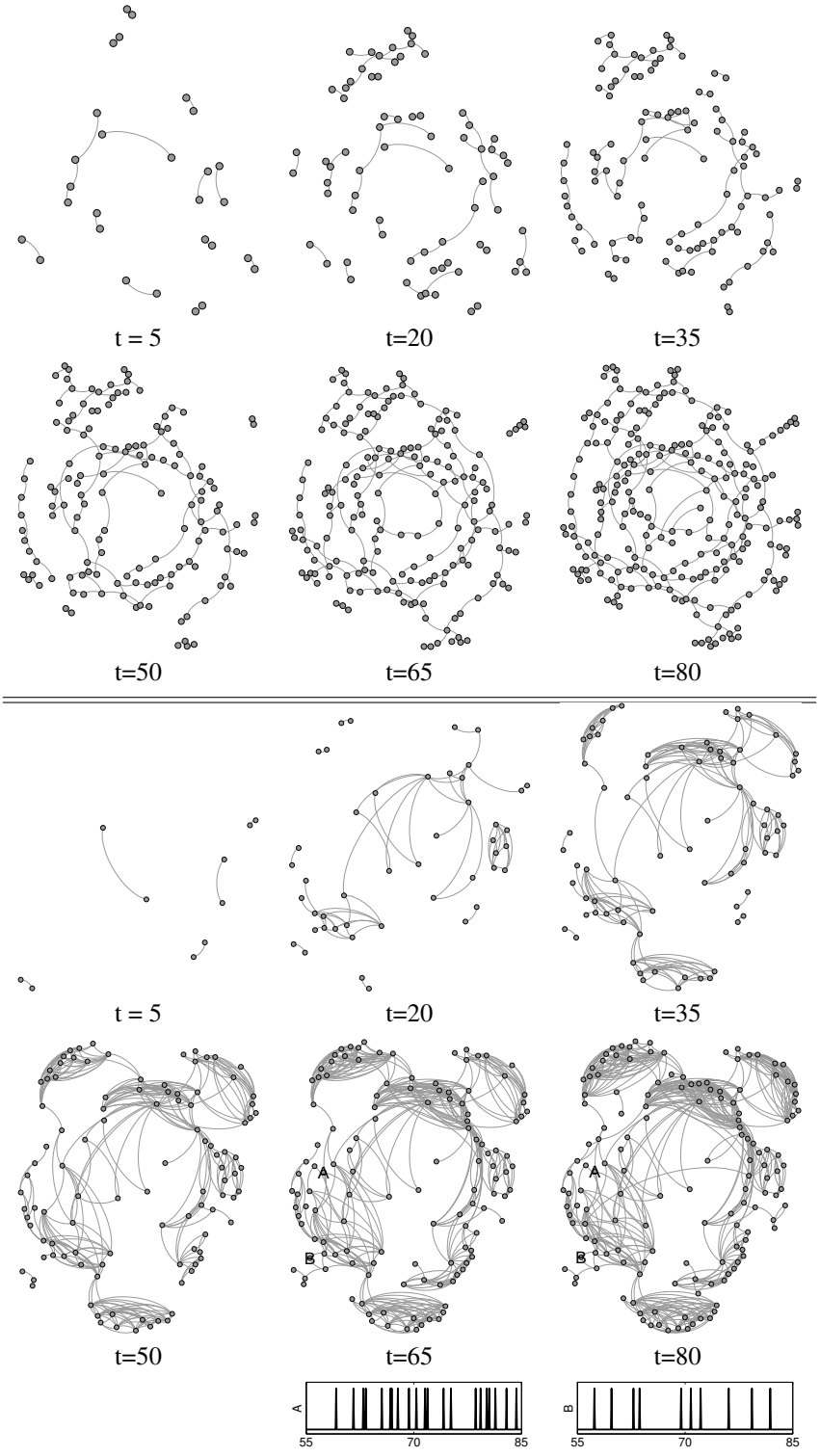


Figure 5: Evolution of a network with $\beta = 0$ (1st and 2nd rows) and $\beta = 0.8$ (3rd and 4th rows) and spike trains of nodes A and B (5th row).

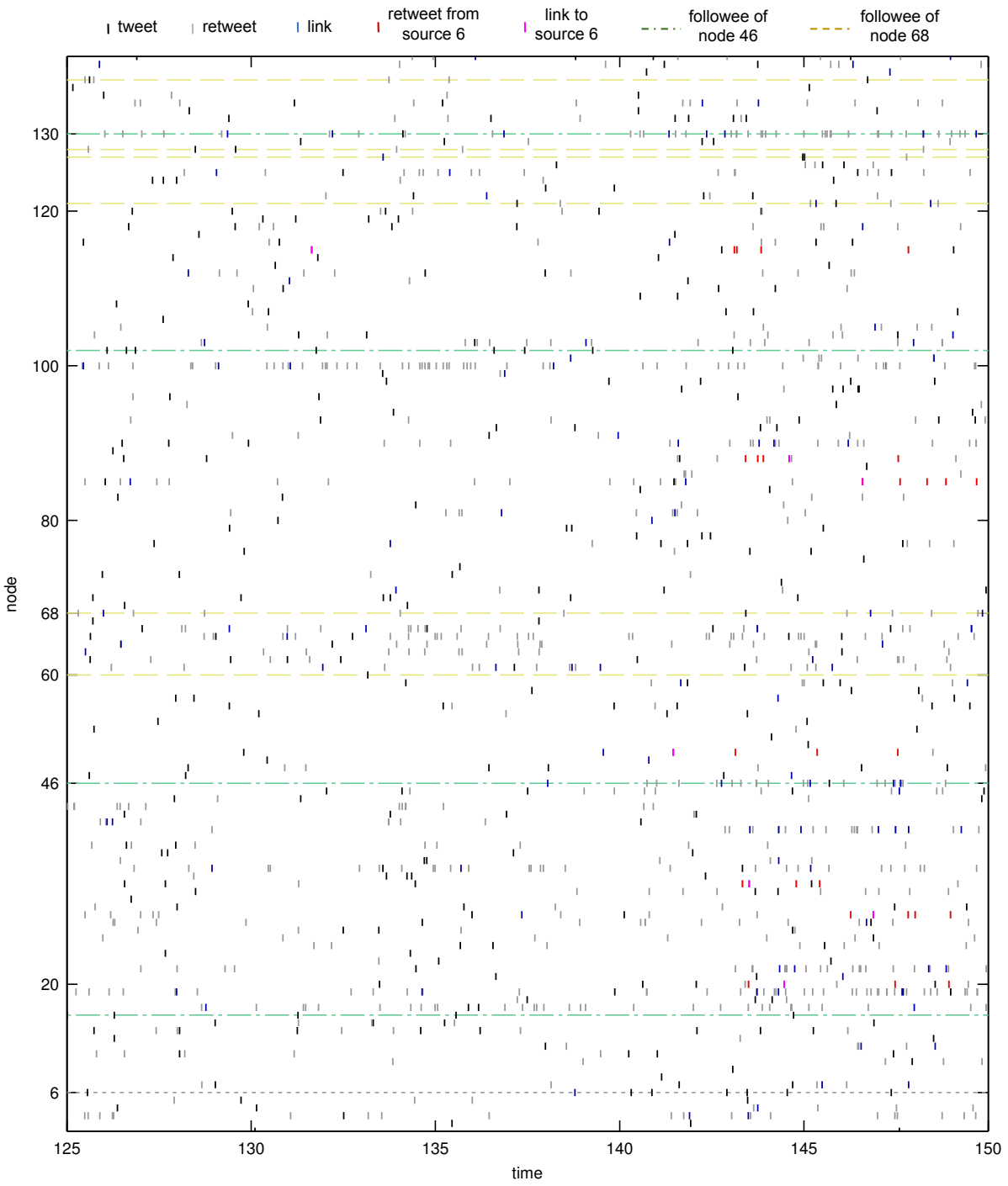


Figure 6: Coevolutionary dynamics of activities and links. The spike trains are associated to networks in Figure 7.
Information Diffusion $\longrightarrow$ Network Evolution: When node 6 joined the network a few nodes followed her and retweeted her tweets. Her tweets being propagated (shown in red) turning her to a valuable source of information. Therefore, those retweets are followed by links created to her (shown in magenta).
Network Evolution $\longrightarrow$ Information Diffusion: Node 46 and 68 both had almost the same number of followees. However, as soon as node 46 connected to node 130 (which is a central node and is retweeintg very much) her activity dramatically increased compared to node 68

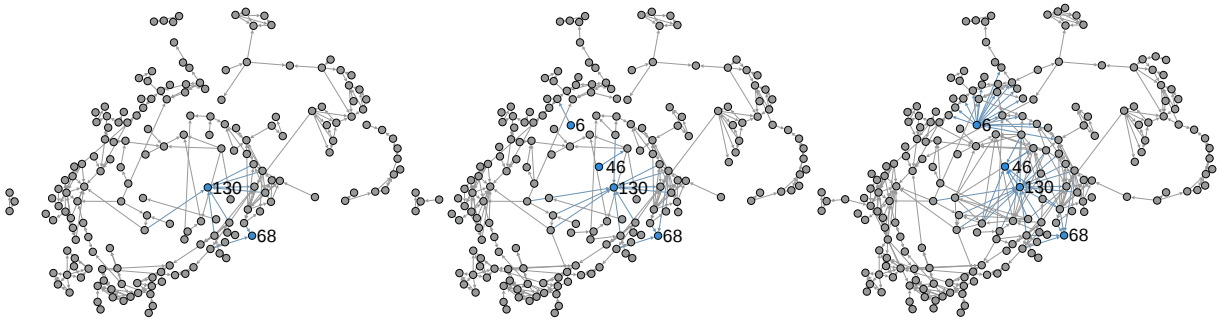


Figure 7: Bottom: Network structure at different times, left: $t = 125$, middle: $t = 137$, right $t = 50$.

are more central, therefore, A is being exposed to more retweets. As an expected consequence node A will perform more retweets than B does.

Figure 6 illustrates the spike trains (tweet, retweet, and link events) for the first 140 nodes of a network simulated with a similar set of parameters as above and Figure 7 shows three snapshots of the network at different times. First, consider node 6 in the network. It joins the network around lately. A few nodes begin to follow him. Suddenly she starts to tweet and her tweets has been retweeted many times by others as illustrated via red spikes in the figure. It's easily seen that shortly after being retweeted she has been found as a valuable source of information. Links established to her is illustrated by Magenta spikes which usually come after red ones. This clearly indicates how the information diffusion affects network structure. On the other side, let's see how the framework models the influence of network structure on information diffusion. Consider nodes 46 and 68 and compare their activities. Node 46 has been more active lately and performs many retweets. To see why, note that soon after time 137 node 46 follows node 130 which is a very central node (*i.e.* following a lot of people) from information point of view. However, node 68 does not connect to such active users. This clearly explains why node 46 becomes very active as he has been exposed to much information.

6.7 Cascade Patterns

Our model can produce the most commonly occurring cascades structures as well as heavy-tailed cascade size and depth distributions, as observed in historical Twitter data [25]. Figure 8 summarizes the results. The higher the α (β) value, the shallower and wider the cascades.

7 Experiments on Model Estimation and Prediction on Synthetic Data

In this section, we first show that our model estimation method can accurately recover the true model parameters from historical link and diffusion events data and then demonstrate that our model can accurately predict the network evolution and information diffusion over time, significantly outperforming two state of the art methods [5, 3, 4] at predicting new links, and a baseline Hawkes process that does not consider network evolution at predicting new diffusion events.

7.1 Experimental Setup

Throughout this section, we experiment with our model considering m=400 nodes. We set the model parameters for each node in the network by drawing samples from $\mu \sim U(0, 0.0004)$, $\alpha \sim U(0, 0.1)$, $\eta \sim U(0, 1.5)$ and $\beta \sim U(0, 0.1)$. We then sample up to 60,000 link and information diffusion events from our model using Algorithm 1 and average over 8 different simulation runs.

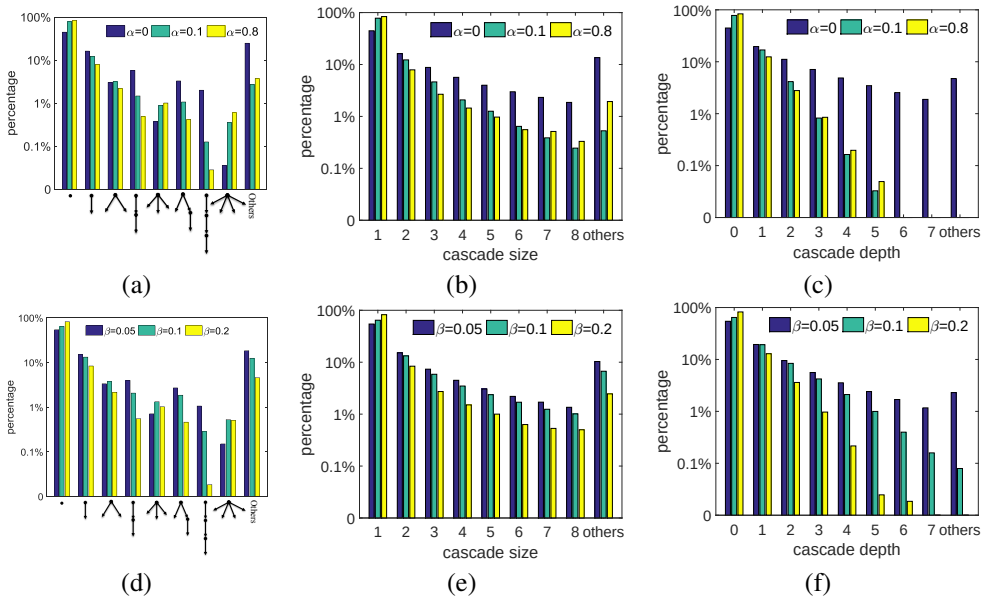


Figure 8: Distribution of cascade structure, size and depth for different α (β) values and fixed $\beta = 0.2$ ($\alpha = 0.8$).

7.2 Model Estimation

We evaluate the accuracy of our model estimation procedure via two measures: (i) the relative mean absolute error (*i.e.*, $\mathbb{E}[|x - \hat{x}|/x]$, MAE) between the estimated parameters (x) and the true parameters ($\hat{x}$), (ii) the Kendall's rank correlation coefficient between each estimated parameter and its true value, and (iii) test log-likelihood. Figure 9 shows that as we feed more events into the estimation procedure, the estimation becomes more accurate.

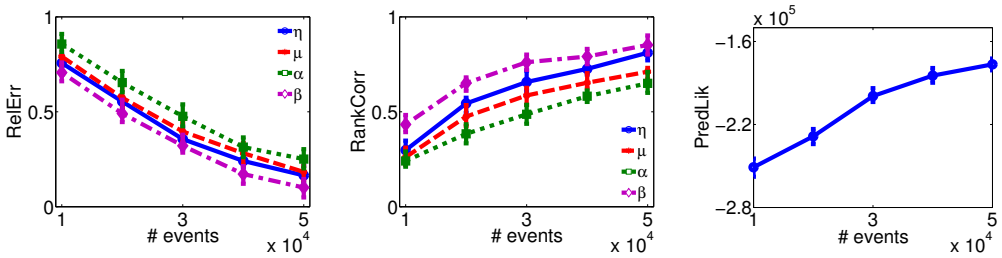
7.3 Link Prediction

We use our model to predict the identity of the source for each test link event, given the historical events before the time of the prediction, and compare its performance with two state of the art methods, which we denote as TRF [3] and WENG [5]. TRF measures the probability of creating a link from a source at a given time by simply computing the proportion of new links created from the source over all total created links up to the given time. WENG considers different link creation strategies and makes a prediction by combining these strategies.

Here, we evaluate the performance by computing the probability of all potential links using our model, TRF and WENG and then compute (i) the average rank of all true (test) events (AvgRank) and, (ii) the success probability that the true (test) events rank among the top-1 potential events at each test time (Top-1). Figure 10 summarizes the results, where we trained our model with an increasing number of events. Our model outperforms both TRF and WENG for a significant margin.

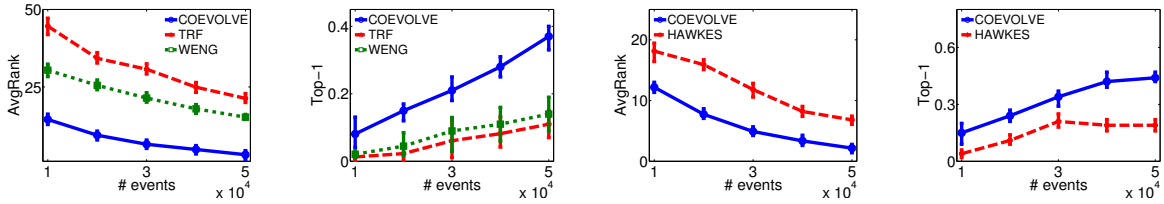
7.4 Activity Prediction

We use our model to predict the identity of the node that is going to generate each test diffusion event, given the historical events before the time of the prediction, and compare its performance with a baseline consisting of a Hawkes process without network evolution. For the Hawkes baseline, we take a snapshot of the network right before the prediction time, and use all historical retweeting events to fit the model. Here, we evaluate the performance via the same two measures as in the link prediction task and summarize the results in Figure 10 against an increasing number of training events. The results show that, by modeling the network evolution, our model performs significantly better than the baseline.



(a) Relative MAE (b) Rank correlation (c) Test log-likelihood

Figure 9: Performance of model estimation for a 400-node synthetic network.



(a) Links: AR (b) Links: Top-1 (c) Activity: AR Activity: Top-1

Figure 10: Prediction performance for a 400-node synthetic network by means of average rank (AR) and success probability that the true (test) events rank among the top-1 events (Top-1).

8 Experiments on Coevolution and Prediction on Real Data

In this section, we validate our model using a large Twitter dataset containing nearly 8 million tweet, retweet and link events from more than 6 million users [3]. We will show that our model can capture the co-evolutionary dynamics and, by doing so, it predicts retweet and link creation events more accurately than several alternatives.

8.1 Dataset Description & Experimental Setup

We use a dataset that contains both link events as well as tweets/retweets from millions of Twitter users [3]. In particular, the dataset contains data from three set of users. The first set of users (8,779 users) are source nodes s, for whom all their tweet times were collected. The second set of users (77,200 users) are the followers of the first set of users, for whom all their retweet times (and source identities) were collected. The third set of users (6,546,650 users) are the users that start following at least one user in the first set during the recording period, for whom all the link times were collected. In our experiments, we focus on all events (and users) during a 5-day period, in which the users in the first set post 371,012 tweets, the users in the second set retweet 130,275 tweets and the users in the third set create 7,983,306 new link.

We split events into a training set (covering 85% of the retweet and link events) and a test set (covering the remaining 15%) according to time, *i.e.*, all events in the training set occur earlier than those in the test set. We then use our model estimation procedure to fit the parameters from an increasing proportion of events from the training data.

8.2 Retweet and Link Coevolution

Figures 11 visualize the retweet and link events, aggregated across different sources, and the corresponding intensities given by our trained model for four nodes, picked at random. Here, it is already apparent that retweets and link creations are clustered in time and often follow each other, and our fitted model intensities successfully track such behavior. Further, Figure 12 compares the cross-covariance between the empirical retweet and link creation intensities and between the retweet and link creation intensities given by our trained model, computed across multiple realizations, for the same nodes. For all nodes, the similarity between both cross-covariances is striking and both has its peak around

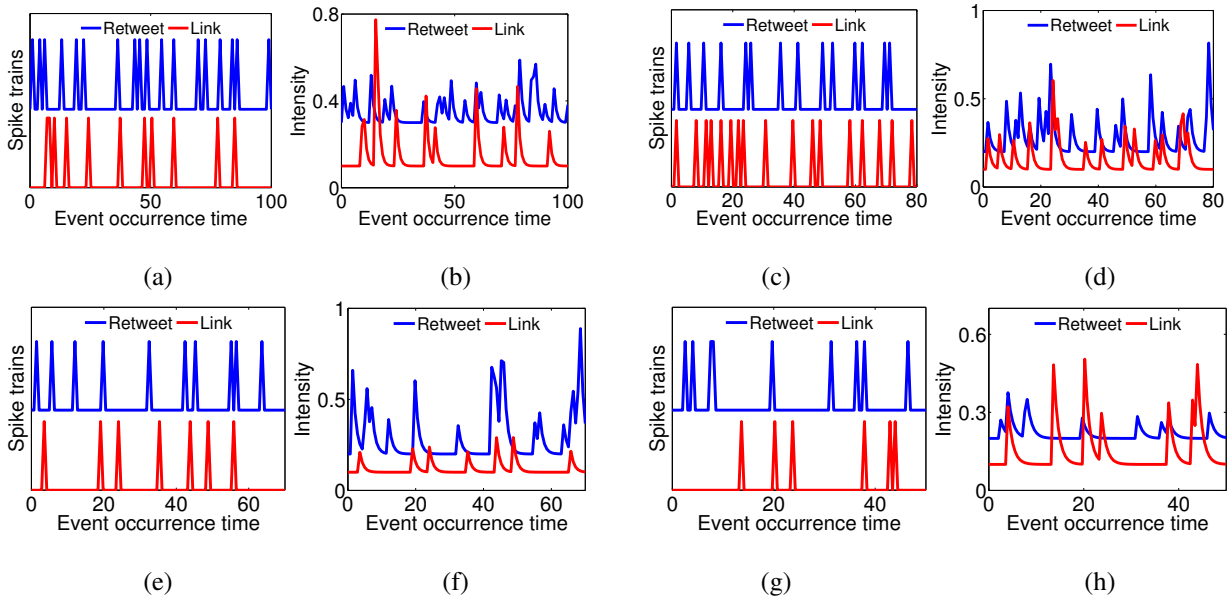


Figure 11: Link and retweet behavior of 4 typical users in the real-world dataset. Panels (a,c,e,g) show the spike trains of link and retweet events and Panels (b,d,f,h) show the estimated link and retweet intensities

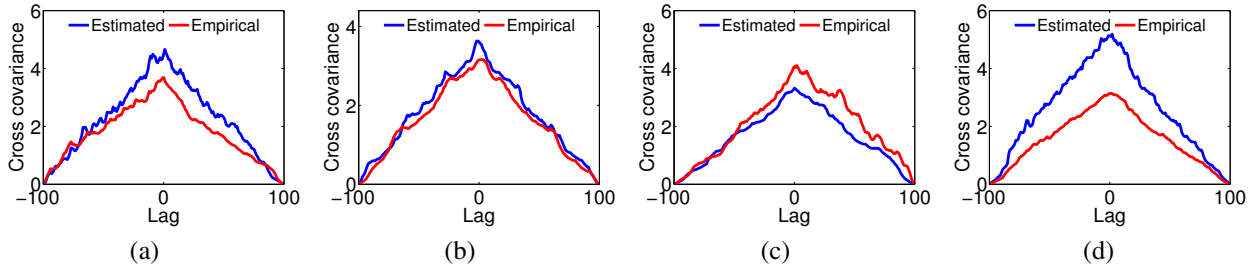


Figure 12: Empirical and simulated cross covariance of link and retweet intensities for 4 typical users.

0, *i.e.*, retweets and link creations are highly correlated and co-evolve over time. For ease of exposition, as in Section 6, we illustrated co-evolution using four nodes, however, we found consistent results across nodes.

To further verify that our model can capture the coevolution, we compute the average value of the empirical cross covariance function, denoted by m_{cc}, per user. Intuitively, one could expect that our model estimation method should assign higher α and/or β values to users with high m_{cc}. Figure 13 confirms this intuition on 1,000 users, picked at random users. Whenever a user has high α and/or β value, she exhibits a high cross covariance between her created links and retweets.

8.3 Link prediction

We use our model to predict the identity of the source for each test link event, given the historical (link and retweet) events before the time of the prediction, and compare its performance with two state of the art methods, denoted as TRF [3] and WENG [5]. TRF measures the probability of creating a link from a source at a given time by simply computing the proportion of new links created from the source with respect to the total number of links created up to the given time. WENG considers different link creation strategies and makes a prediction by combining them.

We evaluate the performance by computing the probability of all potential links using different methods, and then compute (i) the average rank of all true (test) events (AvgRank) and, (ii) the success probability (SP) that the true

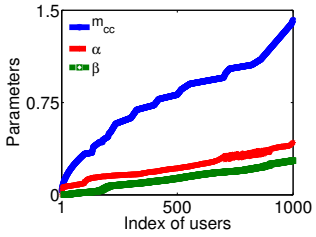
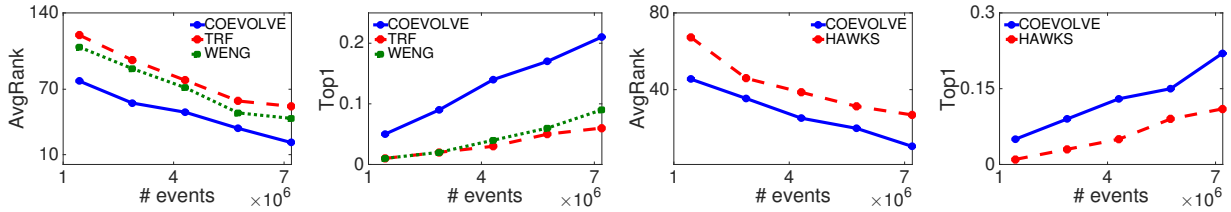


Figure 13: Empirical cross covariance and learned model parameters for 1,000 users, picked at random



(a) Links: AR (b) Links: Top-1 (c) Activity: AR Activity: Top-1

Figure 14: Prediction performance in the Twitter dataset by means of average rank (AR) and success probability that the true (test) events rank among the top-1 events (Top-1).

(test) events rank among the top-1 potential events at each test time (Top-1). We summarize the results in Fig. 14(a-b), where we consider an increasing number of training retweet/tweet events. Our model outperforms TRF and WENG consistently. For example, for $8 \cdot 10^4$ training events, our model achieves a SP 2.5x times larger than TRF and WENG.

8.4 Activity prediction

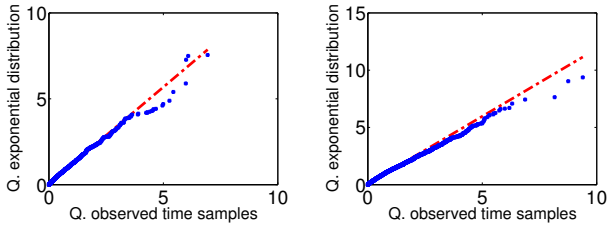
We use our model to predict the identity of the node that is going to generate each test diffusion event, given the historical events before the time of the prediction, and compare its performance with a baseline consisting of a Hawkes process without network evolution. For the Hawkes baseline, we take a snapshot of the network right before the prediction time, and use all historical retweeting events to fit the model. Here, we evaluate the performance the via the same two measures as in the link prediction task and summarize the results in Figure 14(c-d) against an increasing number of training events. The results show that, by modeling the co-evolutionary dynamics, our model performs significantly better than the baseline.

8.5 Model Checking

Given all t_i and t_{i+1} subsequent event times generated using a Hawkes process, then, by the time changing theorem [13], the intensity integrals $\int_{t_i}^{t_{i+1}} \lambda(t)\, dt$ should conform to the unit-rate exponential distribution. Figure 15 presents the quantiles of the intensity integrals computed using the intensities with the parameters estimated from the real Twitter data against the quantiles of the unit-rate exponential distribution. It clearly shows that the points approximately lie on the same line, giving empirical evidence that a Hawkes process is the right model to capture the real dynamics.

9 Discussion

We proposed a joint continuous-time model of information diffusion and network evolution, which can capture the coevolutionary dynamics, mimics the most common static and temporal network patterns observed in real-world networks and information diffusion data, and predicts the network evolution and information diffusion more accurately



(a) Link process (b) Retweet process

Figure 15: Quantile plots of the intensity integrals from the real link and retweet event time

than previous state-of-the-arts. Using point processes to model intertwined events in information networks opens up many interesting future modeling work. Our current model is just a show-case of a rich set of possibilities offered by a point process framework, which have been rarely explored before in large scale social network modeling. For example, we can generalize our model to support link deletion by introducing an intensity matrix $\mathbf{\Xi}^*(t)$ modeling link deletions as survival processes, *i.e.*, $\mathbf{\Xi}^*(t) = (g^*_{us}(t)A_{us}(t))_{u,s\in[m]}$, and then consider the counting process $\boldsymbol{A}(t)$ associated with the adjacency matrix to evolve as $\mathbb{E}[d\boldsymbol{A}(t)|\mathcal{H}^r(t) \cup \mathcal{H}^l(t)] = \mathbf{\Lambda}^*(t)\,dt - \mathbf{\Xi}^*(t)\,dt$. We can consider the number of nodes $m(t)$ varying over time, modeling its intensity function $\mathbb{E}[dm(t)|\mathcal{H}^r(t) \cup \mathcal{H}^l(t)] = \sigma^*_b(t) - \sigma^*_d(t)$, where $\sigma^*_b(t)$ denotes a birth intensity and $\sigma^*_d(t)$ denotes a death intensity. Furthermore, a large and diverse range of point processes can also be used instead in the framework and augment the current model without changing the efficiency of the simulation and the convexity of the parameter estimation, *e.g.*, condition the intensity on additional external features, such as node attributes or edge types, and incorporate features from previous state of the diffusion or network structure.

References

[1] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is Twitter, a social network or a news media? In *Proceedings of the 19th International Conference on World Wide Web*, pages 591–600, New York, NY, USA, 2010. ACM.

[2] Justin Cheng, Lada Adamic, P Alex Dow, Jon Michael Kleinberg, and Jure Leskovec. Can cascades be predicted? In *Proceedings of the 23rd international conference on World wide web*, pages 925–936, 2014.

[3] Demetris Antoniades and Constantine Dovrolis. Co-evolutionary dynamics in social networks: A case study of twitter. *arXiv preprint arXiv:1309.6001*, 2013.

[4] Seth A Myers and Jure Leskovec. The bursty dynamics of the twitter information network. In *23rd International Conference on the World Wide Web*, pages 913–924, 2014.

[5] Lilian Weng, Jacob Ratkiewicz, Nicola Perra, Bruno Gonçalves, Carlos Castillo, Francesco Bonchi, Rossano Schifanella, Filippo Menczer, and Alessandro Flammini. The role of information diffusion in the evolution of social networks. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 356–364. ACM, 2013.

[6] Nan Du, Le Song, Manuel Gomez-Rodriguez, and Hongyuan Zha. Scalable influence estimation in continuous-time diffusion networks. In *Advances in Neural Information Processing Systems 26*, 2013.

[7] Manuel Gomez-Rodriguez, David Balduzzi, and Bernhard Schölkopf. Uncovering the temporal dynamics of diffusion networks. In *Proceedings of the International Conference on Machine Learning*, 2011.

[8] Manuel Gomez-Rodriguez, Jure Leskovec, and Andreas Krause. Inferring networks of diffusion and influence. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1019–1028. ACM, 2010.

[9] Mehrdad Farajtabar, Manuel Gomez-Rodriguez, Nan Du, Mohammad Zamani, Hongyuan Zha, and Le Song. Back to the past: Source identification in diffusion networks from partially observed cascades. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

[10] Deepayan Chakrabarti, Yiping Zhan, and Christos Faloutsos. R-mat: A recursive model for graph mining. *Computer Science Department*, page 541, 2004.

[11] Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos, and Zoubin Ghahramani. Kronecker graphs: An approach to modeling networks. *Journal of Machine Learning Research*, 11(Feb):985–1042, 2010.

[12] Jure Leskovec, Lars Backstrom, Ravi Kumar, and Andrew Tomkins. Microscopic evolution of social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 462–470. ACM, 2008.

[13] Thomas Josef Liniger. *Multivariate Hawkes Processes*. PhD thesis, Swiss Federal Institute of Technology Zurich, 2009.

[14] Charles Blundell, Jeff Beck, and Katherine A Heller. Modelling reciprocating relationships with hawkes processes. In *nips*, 2012.

[15] Mehrdad Farajtabar, Nan Du, Manuel Gomez-Rodriguez, Isabel Valera, Hongyuan Zha, and Le Song. Shaping social activity by incentivizing users. In *Advances in Neural Information Processing Systems (NIPS)*, 2014.

[16] Nan Du, Mehrdad Farajtabar, Amr Ahmed, Alexander J Smola, and Le Song. Dirichlet-hawkes processes with applications to clustering continuous-time document streams. In *KDD*. ACM, 2015.

[17] Tomoharu Iwata, Amar Shah, and Zoubin Ghahramani. Discovering latent influence in online social activities via shared cascade poisson processes. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 266–274. ACM, 2013.

[18] Scott W Linderman and Ryan P Adams. Discovering latent network structure in point process data. *arXiv preprint arXiv:1402.0914*, 2014.

[19] Isabel Valera, Manuel Gomez-Rodriguez, and Krishna Gummadi. Modeling adoption of competing products and conventions in social media. *arXiv preprint arXiv:1406.0516*, 2014.

[20] Ke Zhou, Hongyuan Zha, and Le Song. Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes. In *Artificial Intelligence and Statistics (AISTATS)*, 2013.

[21] Ke Zhou, Hongyuan Zha, and Le Song. Learning triggering kernels for multi-dimensional hawkes processes. In *International Conference on Machine Learning (ICML)*, 2013.

[22] David Hunter, Padhraic Smyth, Duy Q Vu, and Arthur U Asuncion. Dynamic egocentric models for citation networks. In *Proceedings of the 28th International Conference on Machine Learning*, pages 857–864, 2011.

[23] Duy Q Vu, David Hunter, Padhraic Smyth, and Arthur U Asuncion. Continuous-time regression models for longitudinal networks. In *Advances in Neural Information Processing Systems*, pages 2492–2500, 2011.

[24] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 177–187. ACM, 2005.

[25] Sharad Goel, Duncan J Watts, and Daniel G Goldstein. The structure of online diffusion networks. In *Proceedings of the 13th ACM conference on electronic commerce*, pages 623–638, 2012.

[26] Thilo Gross and Bernd Blasius. Adaptive coevolutionary networks: a review. *Journal of The Royal Society Interface*, 5(20):259–271, 2008.

[27] Philipp Singer, Claudia Wagner, and Markus Strohmaier. Factors influencing the co-evolution of social and content networks in online social media. In *Modeling and Mining Ubiquitous Social Media*, pages 40–59. Springer, 2012.

[28] Long Tran, Mehrdad Farajtabar, Le Song, and Hongyuan Zha. Netcodec: Community detection from individual activities. In *SDM*, 2015.

[29] Odd Aalen, Ornulf Borgan, and Hakon Gjessing. *Survival and event history analysis: a process point of view*. Springer, 2008.

[30] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *SIGKDD*, pages 137–146. ACM, 2003.

[31] Yosihiko Ogata. On lewis' simulation method for point processes. *Information Theory, IEEE Transactions on*, 27(1):23–31, 1981.

[32] Paul Erdos and A Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hungar. Acad. Sci*, 5:17–61, 1960.

[33] Lars Backstrom, Paolo Boldi, Marco Rosa, Johan Ugander, and Sebastiano Vigna. Four degrees of separation. In *Proceedings of the 4th Annual ACM Web Science Conference*, pages 33–42, 2012.

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

[35] Daniel Mauricio Romero and Jon Kleinberg. The directed closure process in hybrid social-information networks, with an analysis of link formation on twitter. In *ICWSM*, 2010.

[36] Johan Ugander, Lars Backstrom, and Jon Kleinberg. Subgraph frequencies: Mapping the empirical and extremal geography of large graph collections. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1307–1318. International World Wide Web Conferences Steering Committee, 2013.

[37] Duncan J Watts and Steven H Strogatz. Collective dynamics of small-world networks. *Nature*, 393(6684):440–442, June 1998.