Correlated Cascades: Compete or Cooperate

Ali Zarezade,[*] Ali Khodadadi,[*] Mehrdad Farajtabar,[†] Hamid R. Rabiee,[*] Hongyuan Zha[†]
[*]Sharif University of Technology, Azadi Ave, Tehran, Iran
[†]Georgia Institute of Tech., North Ave NW, Atlanta, GA 30332, United States
{zarezade,khodadadi}@ce.sharif.edu, mehrdad@gatech.edu, rabiee@sharif.edu, zha@cc.gatech.edu

Abstract

In real world social networks, there are multiple cascades which are rarely independent. They usually compete or cooperate with each other. Motivated by the reinforcement theory in sociology we leverage the fact that adoption of a user to any behavior is modeled by the aggregation of behaviors of its neighbors. We use a multidimensional marked Hawkes process to model users product adoption and consequently spread of cascades in social networks. The resulting inference problem is proved to be convex and is solved in parallel by using the barrier method. The advantage of the proposed model is twofold; it models correlated cascades and also learns the latent diffusion network. Experimental results on synthetic and two real datasets gathered from Twitter, URL shortening and music streaming services, illustrate the superior performance of the proposed model over the alternatives.

Introduction

Social networks and virtual communities play a key role in today's life. People share their thoughts, beliefs, opinions, news, and even their locations in social networks and engage in social interactions by commenting, liking, mentioning and following each other. This virtual world is an ideal place for studying social behaviors and spread of cultural norms (Vespignani 2012), contagion of diseases (Barabasi 2015), advertising and marketing (Valera and Rodriguez 2015) and estimating the culprit in malicious diffusions (Farajtabar et al. 2015a). Among them, the study of information diffusion or more generally *dynamics on the network* is of crucial importance and can be used in many applications. The trace of information diffusion, virus or infection spread, rumor propagation, and product adoption is usually called *cascades*.

In conventional studies of diffusion networks, individual cascades are mostly considered in isolation, *i.e.*, independent of each other (Rodriguez et al. 2015). However in realistic situations, they are rarely independent and can be *competitive*, when a URL shortening service become popular the others receive less attention; or *cooperative*, when usage of Google Play Music correlates with that of Youtube due to, for example, simultaneous arrival of new albums (Fig. 1).

Modeling multiple cascades which are correlated to each other is a challenging problem. Considerable work have done to extend basic diffusion models to competitive case (He et al. 2012; Pathak, Banerjee, and Srivastava 2010; Lu, Chen, and Lakshmanan 2015). Meyer *et al.* proposed a probabilistic model for diffusion of competitive or cooperative contagions (Myers and Leskovec 2012). They estimate the probability of a user being infected given a sequence of previously observed contagions. But the main drawback of these models is that they are all discrete time, which limits the flexibility of model. Valera in (Valera and Rodriguez 2015) proposed a continuous time method for modeling competing products but incapable of learning the latent diffusion network and prone to overfitting.

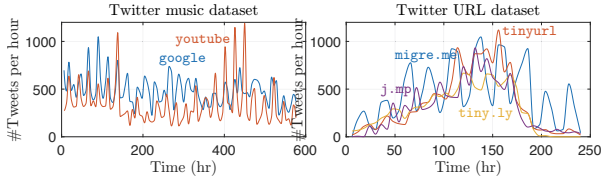

Figure 1: Visualization of correlated cascading behavior in real data. (*left*) Tweets with terms `google` and `youtube` are synchronized most of the time. (*right*) different URL shortening services; `tiny.ly` and `tinyurl` are cooperating while `migre.me` and `j.mp` are competing.

Inspired by the sociological evidence in social science about users' behavior, and the success of the recurrent point processes in modeling temporal event histories, we propose a data-driven continuous time method, which can jointly model the spread of multiple correlated behaviors (information, ideas, memes), and learn the latent diffusion network. Intuitively, the rate (or intensity) that a user adopts a behavior is considered as the weighted sum of those of her neighbors. The more your friends adopt a behavior the more you are excited to adopt it. This users' behavior adoption intensity is modeled by a special stochastic point process, called Hawkes process (Hawkes 1971). It is well suited for modeling temporal events with self-exciting property (Zhou, Zha, and Song 2013; Blundell, Beck, and Heller 2012; Farajtabar et al. 2014; Xu, Farajtabar, and Zha 2016).

Finally, we validate the proposed method on synthetic and two real datasets. First, using synthetic data generated randomly we've studied how effectively we can recover the la-

Copyright © 2017, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

tent diffusion network parameters. We have also highlighted the correlated behavior versus the independent and the competitive versus cooperative cascades using synthetic data. Next, we move forward to real data and show the correlated behavior in the real dataset. Furthermore, with parameters learned from training data, we generate future events and compare it with real held-out test data and show that our framework can model the activities in social network better than the alternatives. Our contributions are as follows:

- Modeling the users' behavior adoption by using a multidimensional marked Hawkes process and consequently, the spread of multiple correlated cascades in social networks.
- Proposing a convex optimization formulation to learn the latent diffusion network and model parameters which is solved in parallel by using the barrier method.
- Curating a compelling dataset on streaming music services using Twitter API, from tweets of 30,000 active users during one month in 2015 year.

Prior Works

The spread of information is often modeled as a dynamical process over networks (Vespignani 2012) and its analysis has attracted significant attention in recent years. These studies can be categorized into three groups.

Early methods studied the information diffusion in continuous time and without any network structure. They are used mainly to analyze biological contagions (Barabasi 2015). The dynamical process is described by a differential equation which models the number of population in different stages of a disease (Porter and Gleeson 2014). Heterogeneous mean-field and particle-network frameworks are proposed to remove the homogeneous population assumption and incorporate the network structure (Vespignani 2012).

Another line of work which is discrete time and considers the network structure, stemmed from sociological theories about influence spread. Typically they assumed that nodes have two states (active, inactive), and are progressive (active nodes can't become inactive). Linear Threshold and Independent Cascade are two simple and widely studied models of social contagion (Hodas and Lerman 2014). In Linear Threshold, a node becomes active when the weighted sum of its active neighbors is higher than a prespecified threshold. In Independent Cascade, each infected node has an independent probability to activate its neighbors.

The third category is continuous time and considers the network structure. Rodriguez et al. proposed a model in (Rodriguez, Balduzzi, and Schölkopf 2011; Rodriguez, Leskovec, and Schölkopf 2013a) for information diffusion and latent influence network inference using survival theory. They extended it to dynamic networks in (Rodriguez, Leskovec, and Schölkopf 2013b). The problem of network inference from a set of cascades is theoretically investigated in (Daneshmand et al. 2014). In (Iwata, Shah, and Ghahramani 2013), the superposition property of the Poisson process is used to model the effect of users' sharing activities on others in online communities, and consequently learning latent influence network. Also in (Linderman and Adams 2014), the superposition property is used in a fully Bayesian method with parallel inference. The scalable influence estimation is addressed in (Du et al. 2013) by proposing a nearly linear randomized algorithm. The problem of activity shaping, driving population toward specific target state is investigated in (Farajtabar et al. 2014; 2016). However, in all of the above models, cascades of adoption/propagation are independent which is usually not true in the real world.

Closely related to the our work, authors in (Farajtabar et al. 2015b) proposed a probabilistic framework to model the evolution of information diffusion and network evolution. However, in their work cascades are still evolving independently. Only recently (Valera and Rodriguez 2015) proposed an algorithm for multiple correlated cascades which models the intensity of user-products by a Hawkes process. To model both competition and cooperation it allows the parameters of the intensity function to be negative, and it may results in negative intensity in some cases. Also, it can't learn the latent diffusion network. But we propose a nonlinear user-product intensity using a marked Hawkes process, which has better performance.

Proposed Method

Hawkes Process Background

A point process is a stochastic process with realizations that are discrete points in time, $\{t_1, t_2, \ldots, t_n\}$. According to the Kolmogorov extension theorem (Daley and Vere-Jones 2002), a stochastic process, can be defined using its finite-dimensional distributions. To describe the finite-dimensional distributions $f(t_1, t_2, \ldots, t_n)$, we use the chain rule of probability: $f(t_1, t_2, \cdots, t_n) = \prod_i f(t_i|t_{1:i-1})$. Therefore, it suffices to describe only the conditionals, which are abbreviated to $f(t_n|\mathcal{H}_n)$ or simply $f^*(t)$. Here, $\mathcal{H}_n$ is the history of events before the n^{th} one. A closely related notion is conditional intensity or rate $\lambda^*(t)$ defined as:

$$\lambda^*(t) = f^*(t)/[1 - F^*(t)], \tag{1}$$

where $F(\cdot)$ is the cdf of $f(\cdot)$. The relation of $\lambda^*(t)$ and $f^*(t)$ can be expressed in the other direction as in (Aalen, Borgan, and Gjessing 2008):

$$f^*(t) = \lambda^*(t) \exp\left(-\int_{t_n}^{t} \lambda^*(s)ds\right).$$

Another basic concept is the survival function, $S^*(t) = 1 - F^*(t)$, the probability that no event occurs after the last event in t_n till t. To understand the intensity more intuitively we incorporate the alternative way of describing a point process, the counting process N associated to $\lambda^*(t)$. Let $N(t, s]$ denotes the number of events in interval $(t, s]$. Multiplying both sides of (1) by dt results in:

$$\begin{aligned}\lambda^*(t)dt &= \frac{\Pr\{N(t_n, t] = 0, N(t, t + dt] = 1|\mathcal{H}_n\}}{\Pr\{N(t_n, t) = 0|\mathcal{H}_n\}} \\ &= \Pr\{N(t, t + dt] = 1|\mathcal{H}_n, N(t_n, t] = 0\} \\ &= \Pr\{N(dt) = 1|\mathcal{H}_{t^-}\} \approx \mathbb{E}\left[N(dt)|\mathcal{H}_{t^-}\right]\end{aligned}$$

where $N(dt) := N(t, t + dt]$ and $\mathcal{H}_{t^-}$ is the history of all events up to t. Different point processes can be constructed

by specifying $f^*(t)$ or equivalently $\lambda^*(t)$. In the Hawkes process the intensity is dependent on the history:

$$\lambda^*(t) = \mu + \int_{-\infty}^{t} g(t-s)N(ds) = \mu + \sum_{i=1}^{|\mathcal{H}_{t^-}|} g(t-t_i)$$

where μ is the base intensity and $g(t)$ is the kernel which is usually exponentially decaying to diminish the effect of past events. Generally, in multidimensional Hawkes process:

$$\boldsymbol{\lambda}^*(t) = \boldsymbol{\mu} + \int_{-\infty}^{t} \mathbf{A} g(t-s)\boldsymbol{N}(ds),$$

where $\boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{N}$ are vectors and $\mathbf{A} = [\alpha_{ij}]$ is a matrix of mutual-excitation kernels. α_{ij} parameterizes the influence of user j to user i. The intensity function can be also generalized to the marked case (Hawkes 1971), which a mark p, often a subset of $\mathbb{N}$ or $\mathbb{R}$, is associated with each event.

$$\lambda^*(t,p) = \lambda^*(t) f^*(p|t),$$

where $f^*(p|t)$ is the conditional mark density function. In the sequel, we omit the star superscript of intensity for notational simplicity. The mutually-exciting property of the Haweks process makes it a common modeling tool in applications like seismology, epidemiology, reliability, and social network analysis (Farajtabar et al. 2015a).

Correlated Cascades Model

Suppose we are given a directed network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $|\mathcal{V}| = N$ nodes and M behaviors (cascades). Nodes of the network can adopt at most one of them in any time. We denote the *user behavior adoption* by $\mathcal{D} = \{(t_i, u_i, p_i)\}_{i=1}^{K}$, where each triple (t_i, u_i, p_i) means that user u_i has adopted behavior p_i at time t_i. We can also define the observations related to user v and behavior (product) q up to time s, as $\mathcal{D}_v^q(s) = \{(t_i, u_i, p_i) \in \mathcal{D} | t_i < s, u_i = v, p_i = q\}$, and define $\mathcal{D}_v(s)$, $\mathcal{D}^q(s)$ and $\mathcal{D}(s)$ in a similar way.

Now the question is, how users in a network decide to adopt a behavior? This is an important question in sociology which has been investigated for decades (Granovetter 1973). According to social reinforcement theory, the behavior of a user is influenced by her friends (McAdam and Paulsen 1993). Moreover, each user has behavioral biases (Farajtabar et al. 2014; 2016). These two mechanisms can be well modeled by the Hawkes process. The mutually-exciting property of the Hawkes process can model the social reinforcement and the base rate can model the bias. Also, the time decaying kernel reflects the diminishing effect of past events. So, we model the behavior adoption intensity of the user u by:

$$\lambda_u(t) = \underbrace{\mu_u}_{\text{bias}} + \underbrace{\sum_{i=1}^{|\mathcal{D}(t)|} \alpha_{u_i u}\, e^{-(t-t_i)}}_{\text{social reinforcement}} \tag{2}$$

where μ_u is the base intensity of user u or bias, α_{ji}, an element of the latent diffusion network, is the influence of user j on i, and the summation is over the elements of the set $\mathcal{D}(t)$. The type of adopted behavior can be seen as the mark of the Hawkes process. Therefore, the intensity of user u to adopt product or behavior p is modeled by: $\lambda_u(t,p) = \lambda_u(t) f_u(p|t)$, where $f_u(p|t)$ is the probability that user u adopts behavior p at time t given history $\mathcal{D}(t)$. To model the mark probability we define the tendency of user u to adopt behavior p as:

$$g_u^p(t) = \mu_u^p + \sum_{i=1}^{|\mathcal{D}^p(t)|} \alpha_{u_i u}\, e^{-(t-t_i)}. \tag{3}$$

Intuitively when a user decides to select a behavior she picks the one with maximum tendency among the different behaviors, $\arg\max_p g_u^p(t)$. The probabilistic version of the max function is the *soft-max* function, so we propose

$$f_u(p|t) = \frac{\exp(\beta g_u^p(t))}{\sum_q \exp(\beta g_u^q(t))}, \tag{4}$$

as the nonlinear mark function, where, hyperparameter β tunes the mark function. In the fully competitive case where $\beta \to \infty$, it converges to deterministic max function, and in the fully cooperative where $\beta \to 0$, it converges to the uniform density function. In the case of linear mark function:

$$f_u(p|t) = \frac{g_u^p(t)}{\sum_q g_u^q(t)} \tag{5}$$

the user behavior intensity simplifies to $\lambda_u(t,p) = g_u^p(t)$. By decomposing the model likelihood to the product of cascade likelihoods, we can show that it reduces to the independent cascade model (Rodriguez, Balduzzi, and Schölkopf 2011). To find the observation likelihood of the proposed model we use the following proposition.

Proposition 1. *For $u = 1, 2, \ldots, N$, let N_u be a multidimensional marked point process on $[0, T]$ with associated intensity $\lambda_u(t)$, and mark density $f_u(p|t)$. Let $\mathcal{D} = \{(t_i, u_i, p_i)\}_{i=1}^{K}$ be a time, user and mark realization of the process over $[0, T]$. Then the likelihood of $\mathcal{D}$ the multidimensional Hawkes process model with mutually-exciting parameter $\boldsymbol{A} = [\alpha_{ij}]$ and baseline parameter $\boldsymbol{\mu} = [\mu_i^p]$, $(i, j = 1, 2, \cdots, N, p = 1, 2, \cdots, M)$ is:*

$$\mathcal{L}(\theta|\mathcal{D}) = \left[\prod_{i=1}^{K} \lambda_{u_i}(t_i) f_{u_i}(p_i|t_i)\right] \exp\left(-\int_0^T \sum_{u=1}^{N} \lambda_u(s) ds\right)$$

where $\theta = (\boldsymbol{\mu}, \boldsymbol{A})$ represents the model parameters.

Proof. Using chain rule, the probability of observation is:

$$\mathcal{L}(\theta|\mathcal{D}) := f(\mathcal{D}|\theta) = \prod_{i=1}^{K} f\left((t_i, u_i, p_i)|\mathcal{D}(t_i)\right) \prod_{u=1}^{N} S(T, u)$$

where $t_0 = 0$ and $S(T, u)$ is the probability that the process $\lambda_u(t)$ survive after its last event:

$$S(T, u) = \exp\left(-\int_{t_{|\mathcal{D}_u|}}^{T} \lambda_u(s) ds\right)$$

by decomposing the probability of observation we have:

$$f(\mathcal{D}|\theta) = \prod_{u=1}^{N} \prod_{i=1}^{|\mathcal{D}_u|} f\left((t_i, u_i, p_i)|\mathcal{D}(t_i)\right) \prod_{u=1}^{N} S(T, u)$$

$$= \prod_{u=1}^{N} \prod_{i=1}^{|\mathcal{D}_u|} \lambda_u(t_i) \exp\left(-\int_{t_{i-1}}^{t_i} \lambda_u(s)ds\right) f_u(p_i|t_i) \prod_{u=1}^{N} S(T,u)$$

$$= \prod_{u=1}^{N} \exp\left(-\int_{0}^{t_{|\mathcal{D}_u|}} \lambda_u(s)ds\right) \prod_{i=1}^{|\mathcal{D}_u|} f_u(p_i|t_i)\lambda_u(t_i) \prod_{u=1}^{N} S(T,u)$$

$$= \prod_{u=1}^{N} \exp\left(-\int_{0}^{t_{|\mathcal{D}_u|}} \lambda_u(s)ds\right) S(T,u) \prod_{i=1}^{|\mathcal{D}_u|} f_u(p_i|t_i)\lambda_u(t_i)$$

$$= \prod_{u=1}^{N} \exp\left(-\int_{0}^{T} \lambda_u(s)ds\right) \prod_{i=1}^{|\mathcal{D}_u|} f_u(p_i|t_i)\lambda_u(t_i)$$

$$= \prod_{u=1}^{N} \exp\left(-\int_{0}^{T} \lambda_u(s)ds\right) \prod_{u=1}^{N} \prod_{i=1}^{|\mathcal{D}_u|} f_u(p_i|t_i)\lambda_u(t_i) \quad \square$$

According to this proposition and relations (2)-(4), the log-likelihood of the model can be written as:

$$\log \mathcal{L}(\theta|\mathcal{D}) = \sum_{i=1}^{|\mathcal{D}|} \log \lambda_{u_i}(t_i) - \sum_{u=1}^{N} \int_0^T \lambda_u(s)ds + \sum_{i=1}^{|\mathcal{D}|} \beta g_{u_i}^{p_i}(t_i) - \sum_{i=1}^{|\mathcal{D}|} \log\left(\sum_{q=1}^{M} \exp\left(\beta g_{u_i}^{q}(t_i)\right)\right)$$

where β is the hyper-parameter. We can decompose the summation over $t_i \in \mathcal{D}$ into the summation over u and $t_i \in \mathcal{D}_u$, which shows that the log-likelihood can be decomposed to sum of users log-likelihood:

$$\log \mathcal{L}(\theta|\mathcal{D}) = \sum_{u=1}^{N} \log \mathcal{L}(\theta_u|\mathcal{D}_u)$$

where the parameters of user u, θ_u is composed of $\boldsymbol{A}_u = [\alpha_{u\cdot}]$ and $\boldsymbol{\mu}_u = [\mu_{u\cdot}]$. Moreover the user's log-likelihood is:

$$\log \mathcal{L}(\theta_u|\mathcal{D}_u) = \sum_{i=1}^{|\mathcal{D}_u|} \log \lambda_u(t_i) - \int_0^T \lambda_u(s)ds + \sum_{i=1}^{|\mathcal{D}_u|} \beta g_{u}^{p_i}(t_i) - \sum_{i=1}^{|\mathcal{D}_u|} \log\left(\sum_{q=1}^{M} \exp\left(\beta g_{u}^{q}(t_i)\right)\right).$$

Lemma 1. *Function $f : \mathbb{R}^n \to \mathbb{R}$ with $\mathrm{dom} f = \mathbb{R}^n$ is convex.*

$$f(x) = \log \sum_i \exp(a_i^T x + b_i)$$

Proof. Let $A = [a_1, a_2, \cdots, a_n]^T$, $b = [b_1, b_2, \cdots, b_n]^T$ and $z_i = \exp(a_i^T x + b_i)$, using chain rule we have:

$$\nabla^2 f(x) = A^T \left(\frac{1}{\mathbf{1}^T z}\mathrm{diag}(z) - \frac{1}{(\mathbf{1}^T z)^2} z z^T\right) A$$

Now we must show that for all u we have $u^T \nabla^2 f(x) u \geq 0$, or equivalently for all v, where $v = Au$ we have:

$$v^T \left(\frac{1}{\mathbf{1}^T z}\mathrm{diag}(z) - \frac{1}{(\mathbf{1}^T z)^2} z z^T\right) v \geq 0$$

$$\frac{\sum_i v_i^2 z_i}{\sum_i z_i} - \left(\frac{\sum_i v_i z_i}{\sum_i z_i}\right)^2 \geq 0$$

which holds according to Cauchy-Schwarz inequality. $\square$

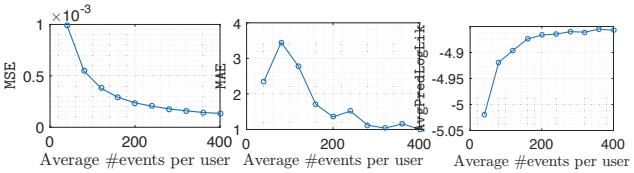

Figure 2: Performance of the parameter learning on synthetic data.

We continue with the following proposition which establishes the tractability of parameter learning and allows us to identify the model efficiently.

Proposition 2. *The negative of the log-likelihood function, $-\log \mathcal{L}(\theta_u|\mathcal{D}_u)$ is convex.*

Proof. The first term is the negative log of a linear function which is convex, according to composition rules. The second and third term are linear, and the fourth term is convex according to lemma 1. $\square$

Now to find the model parameters we can use the maximum likelihood estimation:

$$\underset{\theta}{\text{minimize}} \; -\log \mathcal{L}(\theta_u|\mathcal{D}_u) \qquad \text{subject to } \theta \geq 0$$

on each user, which has unique solution according to proposition 2 and can be solved in parallel for different users.

Experiments[1]

Synthetic Data

We first explain how to generate the synthetic data, then introduce the evaluation criteria. Afterward, we describe the setting for learning the model parameters. Finally, the performance of the algorithm and an experiment that is designed to show the prosperity of the correlated model with respect to its independent version is investigated.

Dataset Preparation. We generated a random network with $N = 50$ and $M = 5$. The parameters of the models were drawn randomly from uniform distribution $\mu_{i,p} \sim U(0, 0.1)$ and $\alpha_{i,j} \sim U(0, 0.01)$. Also, we set $\beta = 1$. Then we sampled 20,000 train events and 2000 test events from the proposed model using the thinning method (Ogata 1981). The convex optimization is solved in parallel using the Barrier method which transforms a constrained convex optimization to an unconstrained one.

Evaluation Criteria. We evaluated the accuracy of learning the model parameters using `MSE`, the average squared error between the estimated and true parameters; `MAE`, the averaged relative error between the estimated and true parameters; and `AvgPredLogLik`, the negative log-likelihood over unseen test events, divided by the number of test events.

Parameter Learning. We trained 10 models, on 10% to 100% of the synthetic training data. In Fig 2, we have evaluated the parameter learning and reported three accuracy measures . To be compatible with the real dataset, we have

[1]Implementation codes and datasets can be found at https://github.com/alikhodadadi/C4.

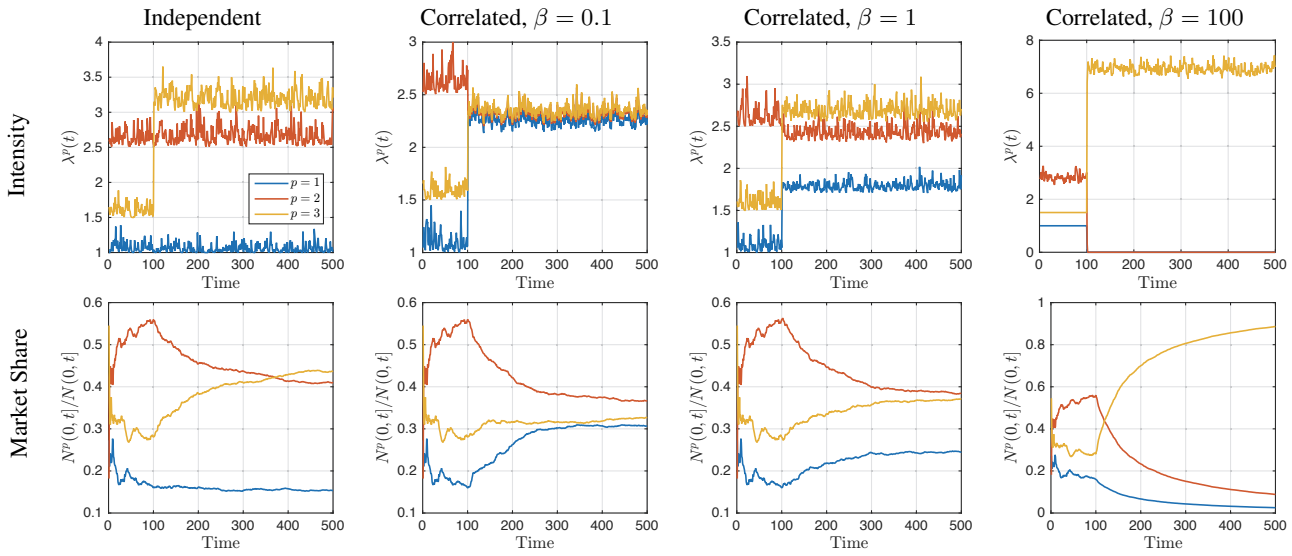

Figure 3: Intensity and market share for independent and different correlated models. In correlated models, after incentivization in time 100, the product usage of the other products also change, whereas in independent models they remain intact.

plotted the measures with respect to the average number of events per user. As expected, with the increase in number of training events the accuracy of recovering the parameters improves.

Correlated Cascades. We also designed an experiment to compare correlated cascade with independent cascade model. We form the independent cascade model using the linear mark (Eq. 5), instead of the exponential mark (Eq. 4). Then randomly generate 4 similar models with the same $\mu_{i,p}$ and $\alpha_{i,j}$, where $\alpha_{i,j} \sim U(0, 0.1)$, and $\mu_{i,p}$ for all users were generated with small noise around 0.2, 0.5 and 0.3 for product $p = 1, 2$ and 3, respectively. The number of nodes and products are also set to $N = 50$, and $M = 3$, respectively. In the correlated models, we set $\beta = 0.1, 1, 100$ to see the effect of mark function on the competitive or cooperative behavior of the proposed model. To show the success of our method in generating the correlated cascades, we design a simple incentivization scenario. For all models, the history of events before time 100 is generated by the independent model. Then the parameters μ_u^p of product $p = 3$ for all users is doubled, which can be regarded as an incentivization of users by the third service provider. Afterward, each model generates its events separately. In Fig. 3 the overall intensity of all users for each product, $\lambda^p(t)$ and the cumulative market share of each product, $N^p(0,t]/\sum_q N^q(0,t]$ is illustrated. From the intensity diagram we can see that, after the incentivization, the intensity of users in the correlated model with $\beta = 0.1$, becomes approximately the same. But in highly competitive model with $\beta = 100$, the third product is dominated shortly after the incentivization which can be seen also from the market share diagrams of Fig 3. To better understand differences between the independent and correlated model, note the intensity of independent model, row and column one of Fig 3. In independent model, by incentivizing product $p = 3$, its intensity increases but the other two product are not influenced by this change. Also the adoption of incentivized product is increased, but the intensity diagram clearly shows that the other two product are not affected by this change. On the contrary in the correlated models, this change affects on the usage of all other products, which validates the correlated nature of our model.

Real Data

In this section, we introduce the real datasets, then explain the evaluation criteria and the settings for parameter learning. Finally, we present the results and the comparisons.

Datasets Preparation. We use the data crawled from Twitter (Hodas and Lerman 2014). This dataset is composed of 213K tweets which contain URLs that shortened by URL shoehorning services. The data was collected over three weeks in Fall of 2010 and is comprised of almost 2K distinct URLs. We post-process this dataset by first finding the six most popular ULR shortening services, which are `bit.ly`, `migre.me`, `tinyurl.com`, `tiny.ly`, `j.mp`, and `is.gd`. Then, we select a collection of tweets of about 1000 users with at least 100 tweets which contain any of the mentioned URLs. We refer to this dataset by "Twitter URL dataset". We have also gathered our own dataset from Twitter. To select a set of active users, we query the Twitter search API, during one week in 2015, with some keywords about recent top music and singers. We select 30,000 users, that were actively tweeting about music and new albums. Then all tweets of these users were crawled using Twitter API, during one month of 2015. To prune this dataset, the tweets containing the URLs of two popular media streaming services, Google Play Music and YouTube are retained. Then, we selected active users with more than 50 tweets. We refer to this dataset by "Twitter music dataset". The intensity (number of tweets per hour) of URL and music datasets are plotted in Fig. 1 in which competition and cooperation be-

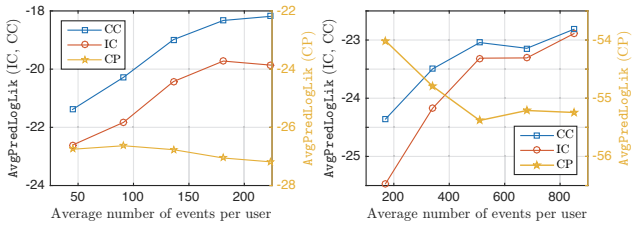

Figure 4: Performance of parameter learning reported via average negative log likelihoods on real datasets, for different size of training set, in Twitter (*left*) URL and (*right*) music datasets. CP is overfitted and the generalization power of the proposed method is more than IC.

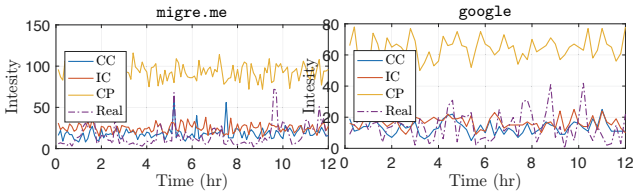

Figure 5: Intensity of generated test events by three methods, compared with real test events in two exemplar products.

tween different cascades is apparent.

Evaluation criteria. We evaluate our model in comparison with two other multiple cascade models. The names are abbreviated by CC, IC, and CP, respectively for Correlated Cascade, Independent Cascade and Competing Products (Valera and Rodriguez 2015) models. In contrast to synthetic data, there is no ground-truth available for real datasets. Hence we use the AvgPredLogLik, Pearson correlation and ℓ_1 distance which measure the prediction accuracy of the model.

Parameter Learning. We set aside the last 20% of the data for the test set. The models are trained five times with 20% to 100% of the train data and β found by cross-validation. The test likelihood for different models is plotted versus the training set size in Fig. 4. The proposed method has the highest likelihood in both datasets and is increasing with respect to the size of the training set. The weak performance of CP is due to overfitting on the training data since the number of parameters in CP is proportional to the square of the number of products. Therefore, the overfit in music dataset (with 2 products) should be more severe than the URL dataset (with 6 products), which can be seen from Fig. 4. The slight decrease in the performance of the proposed method in music dataset is due to mix competitive-cooperative nature of this dataset. As illustrated in Fig. 1, there is a broad range of correlation between cascades. But even in this case, our model has better performance than IC.

Test Events Correlation. To further investigate the proposed method, we also design some experiments on the simulated test events. Using the parameters of the learned model on the whole training data, we generate test events for each model. The model that has higher correlation with real test event, is more successful to predict future events. We exam-

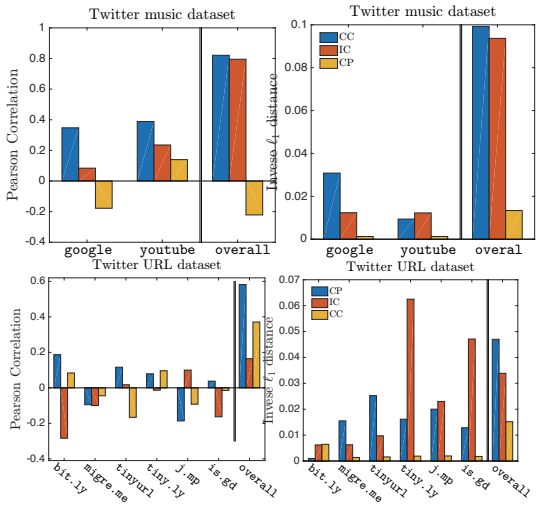

Figure 6: Pearson correlation and inverse ℓ_1 distance for the correlation of simulated test events.

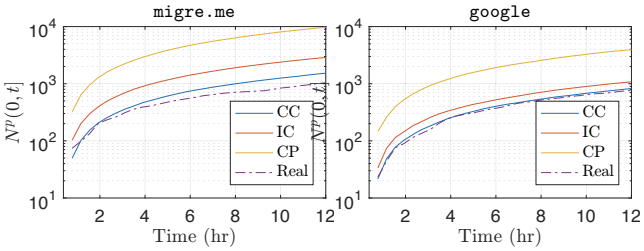

Figure 7: Number of events for the two real exemplar products of Fig 5 .

ined this feature, qualitatively and quantitatively in diagrams of Figs 5 and 6. We show only the intensity of one exemplar product for each dataset in Fig 5. Qualitatively it can be seen that the proposed method better followed the real test intensity curve, except in a few intervals like the times near 2, and 6 hours in the left of Fig 5, that real test intensity has large oscillations. Similar to the poor performance of CP in likelihood on test data, Fig 4, this model has generated more events, which results in its large distance with the true curve. Measuring the distance between two curves is a challenging problem in itself. We use two simple measures, the inverse of ℓ_1 distance, and the well-know Pearson correlation. High inverse ℓ_1 distance and Pearson correlation, indicated a high correlation between the two curves. In Fig 6 the performance of different methods on the two datasets is demonstrated. The result is plotted for two cases; separate products, and all products. In total, we have a higher correlation with the real test events. But like before, the performance of the proposed method in music dataset is slightly better than URL dataset, which is explained already. The number of test events for the mentioned exemplar product of Fig. 5 is depicted in Fig. 7. We use the semi-logarithmic scale in y-axis to better compare different methods. In both cases CC model results are the closest to the real test data.

Conclusion

In this paper, we proposed a social *behavior adoption* model in which multiple correlated cascades spread over the network. Multidimensional Hawkes process is utilized for the behavior or product adoption with its marks capturing the decision making procedure of the users. We have shown several properties of the proposed model on synthetic data. Furthermore, experiments on two real-world datasets establish the competitive-cooperative modeling capability and the superior performance of our model on predicting future events. Importantly, the parameter learning algorithm is shown to be quite efficient in both synthetic and real data.

For future work we would like to learn the hyperparameter and the decaying kernel. Another interesting line of future work would be proposing a model to capture multiple cascades with mixed competing-cooperating behaviors.

References

Aalen, O.; Borgan, O.; and Gjessing, H. 2008. *Survival and event history analysis: a process point of view*. Springer Science & Business Media.

Barabasi, A. L. 2015. *Network Science*. Cambridge university press.

Blundell, C.; Beck, J.; and Heller, K. A. 2012. Modelling reciprocating relationships with hawkes processes. In *Advances in Neural Information Processing Systems (NIPS)*, 2600–2608.

Daley, D., and Vere-Jones, D. 2002. *An Introduction to the Theory of Point Processes - Vol. I*. Springer Ser. Statist., Springer, New York.

Daneshmand, H.; Rodriguez, M. G.; Song, L.; and Schoelkopf, B. 2014. Estimating diffusion network structures: Recovery conditions, sample complexity & soft-thresholding algorithm. In *Proceedings of the... International Conference on Machine Learning. International Conference on Machine Learning*, volume 2014, 793.

Du, N.; Song, L.; Rodriguez, M. G.; and Zha, H. 2013. Scalable influence estimation in continuous-time diffusion networks. In *Advances in Neural Information Processing Systems*, 3147–3155.

Farajtabar, M.; Du, N.; Rodriguez, M. G.; Valera, I.; Zha, H.; and Song, L. 2014. Shaping social activity by incentivizing users. In *Advances in neural information processing systems*, 2474–2482.

Farajtabar, M.; Gomez Rodriguez, M.; Zamani, M.; Du, N.; Zha, H.; and Song, L. 2015a. {Back to the Past: Source Identification in Diffusion Networks from Partially Observed Cascades}. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, 232–240.

Farajtabar, M.; Wang, Y.; Gomez-Rodriguez, M.; Li, S.; Zha, H.; and Song, L. 2015b. Coevolve: A joint point process model for information diffusion and network co-evolution. In *Advances in Neural Information Processing Systems*, 1954–1962.

Farajtabar, M.; Ye, X.; Harati, S.; Song, L.; and Zha, H. 2016. Multistage campaigning in social networks. *arXiv preprint arXiv:1606.03816*.

Granovetter, M. S. 1973. The strength of weak ties. *American journal of sociology* 1360–1380.

Hawkes, A. G. 1971. Spectra of some self-exciting and mutually exciting point processes. *Biometrika* 58(1):83–90.

He, X.; Song, G.; Chen, W.; and Jiang, Q. 2012. Influence blocking maximization in social networks under the competitive linear threshold model. In *SDM*, 463–474. SIAM.

Hodas, N. O., and Lerman, K. 2014. The simple rules of social contagion. *Scientific reports* 4.

Iwata, T.; Shah, A.; and Ghahramani, Z. 2013. 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*, 266–274. ACM.

Linderman, S. W., and Adams, R. P. 2014. Discovering latent network structure in point process data. *arXiv preprint arXiv:1402.0914*.

Lu, W.; Chen, W.; and Lakshmanan, L. V. 2015. From competition to complementarity: Comparative influence diffusion and maximization. *arXiv preprint arXiv:1507.00317*.

McAdam, D., and Paulsen, R. 1993. Specifying the relationship between social ties and activism. *American journal of sociology* 640–667.

Myers, S. A., and Leskovec, J. 2012. Clash of the contagions: Cooperation and competition in information diffusion. In *Data Mining (ICDM), 2012 IEEE 12th International Conference on*, 539–548. IEEE.

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

Pathak, N.; Banerjee, A.; and Srivastava, J. 2010. A generalized linear threshold model for multiple cascades. In *International Conference on Data Mining (ICDM'10)*, 965–970. IEEE.

Porter, M. A., and Gleeson, J. P. 2014. Dynamical systems on networks: A tutorial. *arXiv preprint arXiv:1403.7663*.

Rodriguez, M. G.; Balduzzi, D.; and Schölkopf, B. 2011. Uncovering the temporal dynamics of diffusion networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML'11)*, 561–568.

Rodriguez, M. G.; Song, L.; Daneshmand, H.; and Schoelkopf, B. 2015. Estimating diffusion networks: Recovery conditions, sample complexity & soft-thresholding algorithm. *Journal of Machine Learning Research*.

Rodriguez, M. G.; Leskovec, J.; and Schölkopf, B. 2013a. Modeling information propagation with survival theory. In *Proceedings of The 30th International Conference on Machine Learning (ICML'13)*, 666–674.

Rodriguez, M. G.; Leskovec, J.; and Schölkopf, B. 2013b. Structure and dynamics of information pathways in online media. In *Proceedings of the sixth ACM international conference on Web search and data mining*, 23–32. ACM.

Valera, I., and Rodriguez, M. G. 2015. Modeling adoption and usage of competing products. In *Data Mining (ICDM), 2015 IEEE International Conference on*, 409–418. IEEE.

Vespignani, A. 2012. Modelling dynamical processes in complex socio-technical systems. *Nature Physics* 8(1):32–39.

Xu, H.; Farajtabar, M.; and Zha, H. 2016. Learning granger causality for hawkes processes. *arXiv preprint arXiv:1602.04511*.

Zhou, K.; Zha, H.; and Song, L. 2013. Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics (AISTAT'13)*, 641–649.