Modeling user and topic interactions in social networks using Hawkes processes

Julio Cesar Louzada Pinto
Institut Mines-Telecom
Telecom SudParis
UMR CNRS 5157, France
julio.louzada_pinto@telecom-sudparis.eu

Tijani Chahed
Institut Mines-Telecom
Telecom SudParis
UMR CNRS 5157, France
tijani.chahed@telecom-sudparis.eu

ABSTRACT

We present in this paper a framework to model information diffusion in social networks based on linear multivariate Hawkes processes. Our model exploits the effective broadcasting times of information by users, which guarantees a more realistic view of the information diffusion process. The proposed model takes into consideration not only interactions between users but also interactions between topics, which provides a deeper analysis of influences in social networks. We provide an estimation algorithm based on nonnegative matrix factorization techniques, which together with a dimensionality reduction argument is able to discover, in addition, the latent community structure of the social network. We also provide several numerical results of our method.

Categories and Subject Descriptors

L.6 [**Science and Technology of Learning**]: Learning networks

General Terms

Algorithms, Theory

Keywords

Social networks, Hawkes processes, nonnegative matrix factorization

1. INTRODUCTION

There has been a steady increase of interest in point processes for modeling information diffusion in networks (see [16, 25, 29, 31, 32]). Information diffusion is a phenomenon in social networks where users broadcast information to others in the network; for example on Twitter, users can ”tweet”. By tweeting, users broadcast information to the network; however only those capable of receiving these tweets can retrieve the information, i.e., you must follow the user in question to be able to see his tweets. These sequence of broadcasts by users is called an information cascade.

Following this principle, in this paper we model the information diffusion in a social network by a linear multivariate Hawkes process (see [10, 17]). A Hawkes process is a point process that increases its intensity when an event occurs, hence allowing one to decouple two very different phenomena: the information diffusion due to the willingness of users in propagating their information and the viral network effect of receiving the information from neighbours and retransmitting it; one can think of a group of people participating in a webchat: even though everyone has an intrinsic willingness to discuss, post and interact, if at a given time someone posts a comment over a subject, this comment increases the chance that others will also comment over the same subject, and so on (see for example [4]).

A different reason for the use of point processes in information diffusion models is that they take into consideration the broadcast times of users, whereas standard information cascade models consider time to be discrete, i.e., time only evolves when events occur.

Our motivation comes from information cascade models (see [7, 8, 19, 20, 26, 30]), where one studies the propagation of information in a social network as a cascade of broadcasting by its users. For example: In [29], Yang and Zha study the dissemination of memes in social networks with linear Hawkes processes and couple the point process with a language model in order to estimate the memes. They provide a variational Bayes algorithm for the coupled estimation of the language model, the influence of users and their intrinsic diffusion rates; however, they do not take into consideration the influence that memes may have on one another; moreover, they propose the estimation of the entire social network, not taking into consideration the eventual lack of communication between users. In [19], Myers and Leskovec study the variation of the probability in retransmiting information due to previous exposure to different types of information; they found that, for Twitter, the retransmission probabilities change drastically; however, the approach of Myers and Leskovec does not take into consideration the time between broadcasts of information and the topology of the network. And in [20], Myers *et al.* study the influence of externalities from other nodes on information cascades in networks; they use a point process approach, from which the times of

VALUETOOLS 2014, December 09-11, Bratislava, Slovakia
Copyright © 2015 ICST 978-1-63190-057-0
DOI 10.4108/icst.valuetools.2014.258186

infection are essential for the estimation of parameters, but the topological properties of the network are of secondary concern in their work.

In this paper, we first model information dissemination by linear Hawkes processes using the full information of the broadcast times and we derive not only the influence of users on each other, but also the influence of different types of topics on each other, as in [19, 29].

We follow the ideas in [23, 24] to derive a nonnegative matrix factorization (see [6, 13, 14]) cyclic estimation for the system parameters - the influence of agents, the influence of topics and the intrinsic diffusion rates.

2. LINEAR HAWKES PROCESSES

A multivariate linear Hawkes process (see [10, 17] for more details) is a self-exciting orderly point process $X_t,\ t \in [0, \tau]$, in $\mathbb{R}^R$ with an intensity for the r^{th} coordinate of the form

$$\lambda_t^r = \mu^r + \sum_{r'} \left(\phi_{r,r'} * dX^{r'}\right)_t = \mu^r + \sum_{r'} \int_0^{t-} \phi_{r,r'}(t-s) dX_s^{r'}, \tag{1}$$

where $\phi_{r,r'},\ r, r' \in \{1, 2, \cdots, R\}$ are positive kernel functions responsible for the temporal decaying of all the interactions which happened in the past and $\mu^r \geq 0$ is the intrinsic (baseline) intensity of the point process for the r^{th} coordinate. In our framework, for example, each coordinate X_t^r of the Hawkes process could represent the countnumber of broadcasts by user r until time t.

The use of self-exciting processes here enlightens the necessity of a theory that can model the interaction between people having a conversation or exchanging messages: imagine two people messaging each other through SMS. Normally each one would have its own rhythm of messaging (this is modelled by the intrinsic rate μ in Eqn. (1)), but due to the self-excitation among these people, they will text faster than they would normally do, in order to answer each other; one can model this effect with the temporal kernel $\phi_{r,r'}$ in Eqn. (1).

By orderly, we mean that almost surely X_t does not have more than one jump occurring simultaneously (see [5] for a more formal definition). By the standard theory of point processes (see [5]) we have that an orderly point process is completely characterized by its intensity, which in this case is also a stochastic process.

Throughout this paper we consider kernels of the form

$$\phi_{r,r'}(t) = K_{r,r'}\phi(t), \tag{2}$$

where $K_{r,r'} \geq 0$ are entries of the $R \times R$ interaction matrix K, which represents the interaction of the coordinate r and the coordinate r'. In other words, the kernels $\phi_{r,r'}$ are all of the same time-decaying function and the interactions between different coordinates differ only in intensity, not in type.

Remark: Two very common time-decaying functions are $\phi(t) = \omega e^{-\omega t}.\mathbb{I}_{\{t>0\}}$ a light-tailed exponential kernel (see [23, 24, 29]) and $\phi(t) = (b-1)(a+t)^{-b}.\mathbb{I}_{\{t>0\}}$ a heavy-tailed power-law kernel (see [4]). Although not mentioned here, expectation-minimization (EM) algorithms can be derived in order to estimate the parameters ω in the exponential case (see [9, 15]) and a, b in the power-law case.

3. INFORMATION DIFFUSION BY LINEAR HAWKES PROCESSES

After a brief introduction to linear Hawkes processes we begin the cornerstone of our paper, *information diffusion*. As mentioned before, information diffusion over social networks is to be interpreted as the broadcasting of messages by users. These messages are assumed to have a specific topic, it can be politics, sports, religion, films, etc...

The broadcasting of messages can be done in various ways, depending on the application: measuring tweets or retweets, checking the history of a conversation in a chat room, etc... but they all have one thing in common: messages are broadcasted by a set V of users in a social network.

In our context, the network will be a generic *social network*, defined as a communication graph $G = (V, E)$, where V is the set of users with cardinality $\sharp V = N$ and E is the edge set, i.e., the set with all the possible communication links between users. We assume this graph to be directed and unweighted, and coded by an inward adjacency matrix A such that $A_{i,j} = 1$ if user j is able to broadcast messages to user i, or $A_{i,j} = 0$ otherwise. If one thinks about Twitter, $A_{i,j} = 1$ means that user i follows user j and receives the news published by user j in his or her timeline.

We also assume a hypothesis that messages are of a specific content or *topic*. Meaning that one can discern the message's major topic and label it within K different topics. These topics could be economics, politics, religion, sports, music, cinema, etc...

In light of these explanations, we model the number of messages broadcasted by users as a linear Hawkes process X_t, where $X_t^{i,k}$ is the cumulated number of messages of topic k broadcasted by user i in the time interval $[0, t]$. In other words, our Hawkes process is a $\mathbb{R}^{N \times K}$ point process.

We define our temporal kernel functions as $\phi_{(i,j),(c,k)}(t)$, which measures the temporal influence of a broadcast of a message about topic c by user j on the broadcast of a message about topic k by user i.

3.1 Intensity

Having explicitely defined the basic assumptions and using the machinery of linear Hawkes processes, we factorize our kernel functions in two parts: the influence of users on other users (given by the $N \times N$ matrix J) and the influence of topics on other topics (given by the $K \times K$ matrix B). We thus have the full form of our kernel functions

$$\phi_{(i,j),(c,k)}(t) = J_{i,j} B_{c,k} \phi(t),$$

where $J_{i,j} \geq 0$, $B_{c,k} \geq 0$ and $\phi(t) \geq 0$ satisfies $||\phi||_1 = \int_0^\infty \phi(t)dt < \infty$. This allows one to use $N^2 + K^2$ parameters instead of N^2K^2.

Given the full form of our kernel functions, here is the final form of our intensities

$$\lambda_t^{i,k} = \mu^{i,k} + \sum_c \sum_j B_{c,k} J_{i,j} \int_0^{t-} \phi(t-s) dX_s^{j,c},$$

which in matrix form can be seen as

$$\lambda_t = \mu + J(\phi * dX)_t B. \qquad (3)$$

As said before, not all users can communicate among themselves. Hence one must take into consideration the inward adjacency matrix A given by the underlying structure on the social network. This is done by the relation

$$A_{i,j} = 0 \Rightarrow J_{i,j} = 0,$$

which gives us

$$\lambda_t^{i,k} = \mu^{i,k} + \sum_c \sum_{j \rightsquigarrow i} B_{c,k} J_{i,j} \int_0^{t-} \phi(t-s) dX_s^{j,c}. \qquad (4)$$

In vector form, we have

$$v(\lambda_t) = v(\mu) + (B^T \otimes J)(\phi * v(dX))_t, \qquad (5)$$

where $v(\lambda)$ is the vectorization of a matrix λ and $\otimes$ is the Kronecker product.

3.2 Parameter estimation by cyclic descent

After defining our model in Eqn. (4), we proceed to the estimation of the user-user influence parameter J, the topic-topic influence parameter B and the intrinsic rates μ, following [23]:

Let $(t_n^{j,c})_n$ be the jump times of the point process $X_t^{j,c}$. Define $\delta_{min} = \min_{(i,k,n') \neq (j,c,n)} |t_{n'}^{i,k} - t_n^{j,c}|$ as the minimum elapsed time between jumps of X_t in $[0, \tau]$ and fix $\delta < \delta_{min}$.

We divide $[0, \tau]$ into $T = \lceil \frac{\tau}{\delta} \rceil$ time bins such that we do not have more than one jump of X_t in each bin, in order to preserve the orderliness property of X_t.

Let Y be the $NK \times T$ matrix such that $Y_{l,t} = \frac{v(X_{t\delta})_l - v(X_{(t-1)\delta})_l}{\delta}$, i.e. the row $l = i + (k-1)N$ of Y contains the jumps of $X_t^{i,k}$ at each time bin $((t-1)\delta, t\delta]$.

Define also the $NK \times T$ matrices $\overline{\lambda}$, $\overline{\mu}$ and $\overline{\phi}$ such that $\overline{\lambda}_{i+(k-1)N,t} = \lambda_{(t-1)\delta}^{i,k}$, $\overline{\mu}_{i+(k-1)N,t} = \mu^{i,k}$, $\overline{\phi}_{i+(k-1)N,t} = (\phi * dX^{i,k})_{(t-1)\delta}$, which gives us

$$\overline{\lambda} = \overline{\mu} + (B^T \otimes J)\overline{\phi}. \qquad (6)$$

3.2.1 Maximum likelihood estimation and nonnegative matrix factorization

With the Hawkes intensity discretized, we proceed to the maximum likelihood estimation of the Hawkes parameters. We begin by showing that maximizing the Riemann-sum approximation of the log-likelihood of X is equivalent to minimizing the Kullback-Leibler (KL) divergence between the jumps of X and the intensity λ.

Lemma 1. *If $\int_0^\tau \log(\lambda_t^{i,k}) dX_t^{i,k}$ and $\int_0^\tau \lambda_t^{i,k} dt$ are approximated by their respective Riemann sums, then maximizing the approximated log-likelihood of X_t in $[0, \tau]$ is equivalent to minimizing*

$$D_{KL}(Y|\overline{\lambda}) = \sum_{l,t} d_{KL}(Y_{l,t}|\overline{\lambda}_{l,t}), \qquad (7)$$

where $d_{KL}(x|y) = x \log(\frac{x}{y}) - x + y$ is the Kullback-Leibler divergence of x and y.

Proof. We have that the log-likelihood of X is given by (see for example [5, 21])

$$\mathcal{L} = \sum_{i,k} \left(\int_0^\tau \log \lambda_t^{i,k} dX_t^{i,k} - \int_0^\tau \lambda_t^{i,k} dt \right).$$

Approximating the integrals in $\mathcal{L}$ by their Riemann sums we get

$$\mathcal{L} \sim \sum_{i,k} \sum_t \left(\log \lambda_{(t-1)\delta}^{i,k} (X_{t\delta}^{i,k} - X_{(t-1)\delta}^{i,k}) - \delta \lambda_{(t-1)\delta}^{i,k} \right),$$

thus maximizing the approximation of $\mathcal{L}$ is equivalent to minimizing

$$-\mathcal{L}/\delta \sim \sum_l \sum_t \left(\overline{\lambda}_{l,t} - Y_{l,t} \log \overline{\lambda}_{l,t} \right).$$

With Y fixed, this is equivalent to minimizing

$$D_{KL}(Y|\overline{\lambda}) = \sum_{l,t} d_{KL}(Y_{l,t}|\overline{\lambda}_{l,t}).$$

□

Following Eqn. (6), we have that $\overline{\lambda}$ is a linear combination of several matrices with positive entries, hence the minimization of Eqn. (7) can be solved by nonnegative matrix factorization (NMF) algorithms (see [6, 14]).

Unfortunately, NMF algorithms are not convex on the ensemble of matrices. Nevertheless, they are convex (due to the convexity of the Kullback-Leibler divergence in this case) on each matrix, given that all others are fixed. It can be shown (see [6, 11, 14]) that estimating each matrix given the rest fixed in a cyclic way produces nonincreasing values for Eqn. (7), thus converging to a local maximum of the approximate log-likelihood.

Due to the overwhelming number of user-user interaction parameters $J_{i,j}$ in real-life social networks (where we have $N \sim 10^8$), we factorize J into FG, such that $F \in \mathcal{M}_{N \times d}(\mathbb{R}_+)$ is a $N \times d$ matrix and $G \in \mathcal{M}_{d \times N}(\mathbb{R}_+)$ is a $d \times N$ matrix, with $d \ll N$. This method is similar to clustering our social network communication graph into different communities (see [12]).

Since a composition of a linear function and a convex function remains convex, we have that

$$D_{KL}(Y|\overline{\lambda}) = D_{KL}(Y|\overline{\mu} + (B^T \otimes FG)\overline{\phi})$$

is still convex as a function of one matrix if the rest remains fixed, and the NMF updates still converge to a local minimum of D_{KL}.

Lemmas 2, 3, 4 and 5 give us the exact form of the NMF multiplicative estimation updates for the Hawkes parameters $J = FG$, B and μ.

3.2.2 Estimation of F

We now proceed to the estimation of the first user-user influence matrix factor F by NMF techniques. It is also extremely desirable to uphold the constraint of $A_{i,j} = 0 \Rightarrow J_{i,j} = (FG)_{i,j} = 0$, i.e., we must estimate F and G such that we keep the communication graph unaltered.

This is a very difficult problem, since the NMF updates destroy this relashionship, and the only other way to do so is to estimate each coordinate separately. Since $A_{i,j} \in \{0, 1\}$, we can circumvent this problem using a convex relaxation of this constraint of the form[1] $\eta \times g(\langle 1 - A, FG\rangle)$, with $g : \mathbb{R}_+ \to \mathbb{R}_+$ a convex function and $\eta \geq 0$ a penalization parameter.

Choosing for example g a linear function we have the following penalization $\eta_F \langle 1 - A, FG\rangle$, with derivative $\nabla_F \eta_F \langle 1 - A, FG\rangle = \eta_F(1 - A)G^T$.

Lemma 2. *Let $\rho = (\mathbb{I} \otimes G)\overline{\phi}$ be an auxiliary $dK \times T$ matrix such that $\overline{\lambda} = \overline{\mu} + (B^T \otimes F)\rho$ and let the $N \times T$ block matrices Y^k, $\overline{\lambda}^k$, $\overline{\mu}^k$ and the $d \times T$ block matrices ρ^k and $\overline{\rho}^k$ such that $Y^k_{i,t} = Y_{i+(k-1)N,t}$, $\overline{\lambda}^k_{i,t} = \overline{\lambda}_{i+(k-1)N,t}$, $\overline{\mu}^k_{i,t} = \overline{\mu}_{i+(k-1)N,t}$, $\rho^k_{i,t} = \rho_{i+(k-1)d,t}$ and $\overline{\rho}^k = \sum_{k'=1}^K B_{k',k}\rho^{k'}$.*

We have the following multiplicative[2] estimates for F:

$$F \leftarrow F \odot \frac{\sum_{k=1}^K \left([\frac{Y^k}{\overline{\lambda}^k}](\overline{\rho}^k)^T\right)}{\sum_{k=1}^K 1(\overline{\rho}^k)^T + \eta_F(1 - A)G^T}, \tag{8}$$

where $\eta_F(1 - A)G^T$, with $\eta_F \geq 0$, is a convex penalization term responsible for $A_{i,j} = 0 \Rightarrow (FG)_{i,j} = 0$, i.e., we do not estimate interactions outside the underlying network structure.

Proof. First of all, we have that $(B^T \otimes FG) = (B^T \otimes F)(\mathbb{I} \otimes G)$, thus $\overline{\lambda} = \overline{\mu} + (B^T \otimes F)(\mathbb{I} \otimes G)\overline{\phi}$.

Let F_i be the rows of F and ρ_t be the columns of ρ, with ρ^k_t the columns of the submatrices ρ^k. Then

$$\begin{aligned}\left((B^T \otimes F)\rho\right)_{i+(k-1)N,t} &= \sum_{k'=1}^K B^T_{k,k'} \langle F_i, \rho^{k'}_t\rangle \\ &= \langle F_i, \sum_{k'=1}^K B_{k',k}\rho^{k'}_t\rangle = (F\overline{\rho}^k)_{it}.\end{aligned}$$

Hence

$$\begin{aligned}D_{KL}(Y|\overline{\lambda}) &= \sum_{j,t} d_{KL}(Y_{jt}|\overline{\lambda}_{jt}) \\ &= \sum_t \sum_i \sum_k d_{KL}(Y_{i+(k-1)N,t}|\overline{\lambda}_{i+(k-1)N,t}) \\ &= \sum_k \left(\sum_t \sum_i d_{KL}(Y^k_{i,t}|\mu^{i,k} + (F\overline{\rho}^k)_{i,t}) \right) \\ &= \sum_k D_{KL}(Y^k|\overline{\mu}^k + F\overline{\rho}^k) = D^F_{KL}(F).\end{aligned}$$

One can see that, since D^F_{KL} is a sum of convex functions, it is still a convex function and we can use the multiplicative update rule for F given by [6, 14]. We have thus the following multiplicative update rule

$$F \leftarrow F \odot \frac{\sum_{k=1}^K \left([\frac{Y^k}{\overline{\lambda}^k}](\overline{\rho}^k)^T\right)}{\sum_{k=1}^K 1(\overline{\rho}^k)^T},$$

Since the penalization term $\eta_F(1 - A)G^T$ has all its entries nonnegative, it is added to the denominator of the NMF updates, as in [6]. Following [6], we can rewrite the multiplicative updates with the linear penalization as Eqn. (8). □

3.2.3 Estimation of G

We now proceed to the estimation of the second user-user influence matrix factor G using the same ideas applied to the estimation of F. Again we must use a convex relaxation of the constraint $A_{i,j} = 0 \Rightarrow J_{i,j} = (FG)_{i,j} = 0$. The derivative of this relaxation with respect to G takes the form $\eta_G F^T(1 - A)$.

Unfortunately, since F anf G act as a product, there is a potential identifiability issue of the form $FG = FMM^{-1}G = \tilde{F}\tilde{G}$ where M is any scaled permutation and the pair $\tilde{F} = FM$, $\tilde{G} = M^{-1}G$ is also a valid factorization of J (see [13, 18, 23]). We deal with this issue normalizing the rows of G to sum to 1 (see [13, 23]). This normalization step involves the resolution of a nonlinear system for each row of G to find the associated Lagrange multipliers.

Our constraint thus becomes $G1 = 1$, for which the Karush-Kuhn-Tucker (KKT) conditions are written in matrix form as $\overline{\eta}_G = \sum_{i=1}^d \eta_{G,i} e_i 1$, with $\eta_{G,i} \in \mathbb{R}$ the Lagrange multipliers solution of the nonlinear equation $G1 = 1$ after the update.

Lemma 3. *Let $\overline{\phi}^k$ be the $N \times T$ matrices such that $\overline{\phi}^k_{i,t} = \overline{\phi}_{i+(k-1)N,t}$ and $\overline{\Phi}^k = \sum_{k'} B_{k'k}\overline{\phi}^{k'}$ be an auxiliary $N \times T$ matrix such that $\overline{\lambda}^k = \overline{\mu}^k + FG\overline{\Phi}^k$.*

We have the following multiplicative updates for G:

$$G \leftarrow G \odot \frac{\sum_{k=1}^K F^T\left([\frac{Y^k}{\overline{\lambda}^k}](\overline{\Phi}^k)^T\right)}{\sum_{k=1}^K F^T 1(\overline{\Phi}^k)^T + \eta_G F^T(1 - A) + \overline{\eta}_G}, \tag{9}$$

where $\overline{\eta}_G$ is a $d \times N$ matrix composed by Lagrange multipliers solution of the nonlinear equation $G1 = 1$ and $\eta_G F^T(1 - A)$, with $\eta_G \geq 0$, is responsible for $A_{i,j} = 0 \Rightarrow (FG)_{i,j} = 0$.

[1] From now on we denote by 1 any vector of matrix with entries equal to 1. The dimension of 1 will be clear in the context.

[2] For two matrices A and B of same dimensions, we denote $\frac{A}{B}$ their entrywise division and $A \odot B$ their entrywise product.

PROOF. Firstly, we have that

$$\begin{aligned}D_{KL}(Y|\overline{\lambda}) &= \sum_{j,t} d_{KL}(Y_{jt}|\overline{\lambda}_{jt}) \\ &= \sum_{i,t,k} d_{KL}(Y^k_{i,t}|\mu^{i,k} + \langle \sum_{k'=1}^{K} B_{k'k}F_i, G\overline{\phi}^{k'}_t \rangle) \\ &= \sum_k \left(\sum_{i,t} d_{KL}(Y^k_{it}|\big(\overline{\mu}^k + FG\overline{\Phi}^k\big)_{i,t}) \right) \\ &= \sum_k D_{KL}(Y^k|\overline{\mu}^k + FG\overline{\Phi}^k) = D^G_{KL}(G).\end{aligned}$$

Using the same arguments as with F, we have the update rule for G given by Eqn. (9). □

3.2.4 Estimation of B

For the estimation of the topic-topic influence matrix B, one may also notice that we still need to normalize the rows of B to sum up to 1, for the same reasons as in G, since B appears multiplying $J = FG$ in Eqn. (5).

LEMMA 4. *Let ζ^i be $K \times T$ matrices such as $\zeta^i_{k,t} = \big(J\overline{\phi}^k\big)_{i,t}$.*

Let $\underline{Y}^i$, $\underline{\mu}^i$ and $\underline{\lambda}^i$ be $K \times T$ matrices such that $\underline{Y}^i_{k,t} = Y_{i+(k-1)N,t}$, $\underline{\mu}^i_{k,t} = \overline{\mu}_{i+(k-1)N,t} = \mu^{i,k}$ and $\underline{\lambda}^i_{k,t} = \overline{\lambda}_{i+(k-1)N,t} = \mu^{i,k} + \big(B^T\zeta^i\big)_{k,t}$.

We have the following multiplicative updates for B:

$$B \leftarrow B \odot \frac{\sum_{i=1}^N \zeta^i [\frac{(\underline{Y}^i)^T}{(\underline{\lambda}^i)^T}]}{\sum_{i=1}^N \zeta^i 1 + \overline{\eta}_B}, \tag{10}$$

where $\overline{\eta}_B$ is a matrix composed by the Lagrange multipliers solution of the nonlinear equation $B1 = 1$.

PROOF. Firstly, we have

$$\begin{aligned}D(Y|\overline{\lambda}) &= \sum_{j,t} d_{KL}(Y_{jt}|\overline{\lambda}_{jt}) \\ &= \sum_{i,t,k} d_{KL}(\underline{Y}^i_{k,t}|\mu^{i,k} + \sum_{k'} B^T_{kk'}\zeta^i_{k',t}) \\ &= \sum_i \left(\sum_{i,t} d_{KL}(\underline{Y}^i_{k,t}|\big(\underline{\mu}^i + B^T\zeta^i\big)_{k,t}) \right) \\ &= \sum_i D_{KL}(\overline{Y}^i|\underline{\mu}^i + B^T\zeta^i) = D^B_{KL}(B^T).\end{aligned}$$

By the same principle as in the estimation of F and G, the updates for B are given by Eqn. (10). □

3.2.5 Estimation of μ

Applying the same techniques, we can estimate the users intrinsic rates matrix μ.

LEMMA 5. *We have the multiplicative updates for μ:*

$$v(\mu) = v(\mu) \odot \frac{[\frac{Y}{\overline{\lambda}}]1}{\langle 1, 1 \rangle} = v(\mu) \odot \frac{[\frac{Y}{\overline{\lambda}}]1}{T}. \tag{11}$$

PROOF. By the same token, it is easy to see that

$$\begin{aligned}D(Y|\overline{\lambda}) &= \sum_{j,t} d_{KL}(Y_{jt}|\big(v(\mu)1 + (B^T \otimes J)\overline{\phi}\big)_{jt}) \\ &= \sum_{j,t} d_{KL}(Y_{jt}|\big(v(\mu)1\big)_{jt} + \big((B^T \otimes J)\overline{\phi}\big)_{jt}) \\ &= D^{\mu}_{KL}(Y|\mu),\end{aligned}$$

giving us the multiplicative updates in Eqn. (11). □

3.3 Complexity

NMF factorization techniques are multiplicative updates, using only entrywise operations and matrix products, which is fast and can be performed in a distributed fashion very easily. Hence, at each step of the cyclic descent procedure, we have the following complexity for the updates, written in terms of the number of users N, the number of topics K, the factorization dimension d and the number of time discretization steps T:

- The complexity of updating F is $\mathcal{O}(dKNT + dKN^2)$.
- The complexity of updating G is $\mathcal{O}(dK^2NT + dKN^2)$.
- The complexity for the numerator and denominator updates of B is $\mathcal{O}(K^2NT)$.
- The complexity for μ is $\mathcal{O}(dKNT)$.

For the complexity of G and B, we also have to take into consideration the calculation of the Lagrange multipliers $\overline{\eta}_G$ and $\overline{\eta}_B$. These multipliers are calculated using convex optimization techniques[3]. However, the complexity of these calculations is not greater than the complexity of the multiplicative updates for G or B.

3.3.1 Total complexity of the updates

The complexity of each cyclic updating step (updating F with the rest fixed, updating G with the rest fixed, updating B with the rest fixed and updating μ with the rest fixed) is thus

$$\mathcal{O}\big(dNK^2T + dN^2K\big) = \mathcal{O}(dNK^2T)$$

if $N \ll T$, which is normally the case (we usually have considerably more messages than users).

Thus, we achieve a linear complexity on the dataset - which is basically dictated by N and T since $K \ll N$, $K \ll T$ and $d \ll N$.

3.3.2 Complexity for J without the factorization FG

Following the same calculations as for the complexity of F using Eqn. (8), we get that the complexity for J is $K \times \mathcal{O}(N^2T) = \mathcal{O}(KN^2T)$.

By the same token, every time we factorize $J = FG$ to compute the other multiplicative updates for B and μ, we have to calculate $\overline{\lambda}$, which has a complexity of $\mathcal{O}(dKNT)$. If we cannot factorize J, the complexity becomes $\mathcal{O}(KN^2T)$, which is much larger than $\mathcal{O}(dKNT)$ since $d \ll N$.

[3] Since we need to find the zero of the function $h(\eta) = \frac{1}{a+\eta}$.

This proves that the dimensionality reduction $J = FG$ is crucial to obtain a linear complexity in the data.

3.4 Additional remarks

Our estimation method - based on the maximum log-likelihood of the point process X_t and on nonnegative matrix factorization techniques - requires the NMF parameter d. This parameter is ad-hoc and must be learned beforehand. However, Tan and Févotte derive an automatic way of finding the optimal d during the NMF updates in [27]. They do so by considering the NMF procedure for the β-divergence (for which the KL divergence is a particular case) as a Bayesian estimation of an underlying probabilistic model.

One known setback in the NMF framework is the convergence to local minima of the cost function, which means that the initial condition is crucial for a good estimation. There are results that illustrate how to achieve a better estimation by constructing an improved initial condition (see [1, 3, 28]), but they do not work here: our cost function is with respect to $D_{KL}(Y|\overline{\mu}+(B^T \otimes FG)\overline{\phi})$ and the frameworks in [1, 3] do not apply if we consider finding good initial conditions for $J = FG$, B and μ *at the same time*. Moreover, we do not know the true value of $J = FG$, our only proxy is the adjacency matrix A, which is binary ($A_{i,j} \in \{0, 1\}$) and make it very hard to use the methods in [3, 28]. We use random initial conditions for B and μ, and we factorize A into $A = F_A G_A$, with $F_A \in \mathcal{M}_{N\times d}(\mathbb{R}^+)$ and $G_A \in \mathcal{M}_{d\times N}(\mathbb{R}^+)$, and use F_A as the initial condition for F and G_A as the initial condition for G.

If estimating parametric kernels ϕ of exponential or power-law type (see section 2), the convolution $\phi * dX$ must be calculated at each NMF update, which increases considerably the running time of the algorithm, since calculating $\phi * dX$ is costlier than the NFM updates. However, for the exponential kernel we could calculate $\phi * dX$ only up to a fixed lag L, as in [29], which speeds up the algorithm.

There are also attempts to derive nonparametric estimation of kernels for Hawkes processes, as in Bacry and Muzy [2]. The problem with nonparametric kernel estimation is the high dimension of our Hawkes processes, i.e., $N \gg 1$ and $T \gg 1$; since these methods are quadrature-based methods for the convolutions, they are much slower than parametric alternatives.

4. NUMERICAL SIMULATION

This section is dedicated to the estimation of our model parameters F, G, B and μ using synthetic Hawkes processes simulated following the thinning algorithm[4] developed by Ogata in [22]. We used for figures 1, 2, 3 and 4 the parameters $N = 100$, $K = 10$ and an exponential temporal kernel of the form $\phi(t) = e^{-10t}.\mathbb{I}_{\{t>0\}}$.

Figures 1, 2, 3 and 4 are from a network composed of two cliques of size 50 with uniform random weights. We simulated our Hawkes process until time $\tau = 250$, and used $d = 51$ for our factorization $J = FG$, with a linear penalization (as in lemma 2) with constants $\eta_F = \eta_G = 10^3$. We did not use cross-validation techniques to find optimal penalization parameters η_F and η_G, since the algorithm is robust enough with respect to them.

Figure 1 is the heatmap of $J = FG$, where the left heatmap is the estimated $J = FG$ and the right heatmap is the true value for J. One can clearly see that our algorithm retrieves quite well the structure behind the true J, i.e., two distinct cliques.

Figure 2 is the heatmap of the squared difference of the true J and its estimation $\tilde{J}$, i.e., for each true entry $J_{i,j}$ and estimated entry $\tilde{J}_{i,j}$ we have ploted the differences $(J_{i,j} - \tilde{J}_{i,j})^2$ and $(J_{i,j} - \tilde{J}_{i,j})^2/J^2_{i,j}$ (when $J_{i,j}$ is nonzero).

Figure 3 refers to the squared difference of B and its estimation and figure 4 refers to the squared difference of the true μ and its estimation, as in figure 2.

Figures 5 is, again, related to a 2-clique network with cliques of size 10, random edge weights following an uniform distribution, $K = 1$ (only one content), and a simulated Hawkes process until $\tau = 20$. We compare our estimation choosing $d = 10$ with the estimation algorithm in [29] (with the obvisouly simplification of $K = 1$ and no language model); one can see that our algorithm (on the left) outperforms the algorithm in [29] not only on the estimation of μ, but also on the estimation of J, retrieving the community structure when the algorithm in [29] did not. Moreover, the algorithm in [29] needs an ad-hoc parameter ρ to control the sparsity of the network, which is not needed in our case.

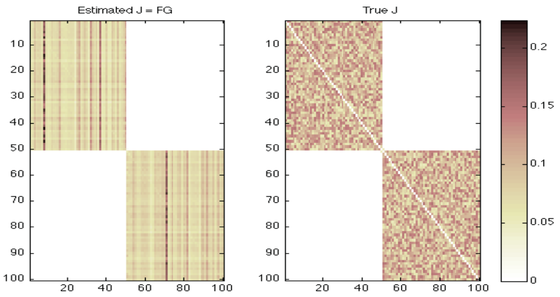

Figure 1: Heatmap of $J = FG$ for 2-clique network of 100 nodes.

5. CONCLUSIONS

We presented in this paper a general framework to model information diffusion in social networks based on the theory of self-exciting point processes - linear multivariate Hawkes processes - and use nonnegative matrix factorization techniques to derive our estimation algorithm.

The model studied here exploits the real broadcasting times of users - a feature that comes with no mathematical overhead since we do so in the framework of point processes (see

[4] The thinning algorithm simulates a standard Poisson process P_t with intensity $M > \sum_{i,k} \lambda_t^{i,k}$ for all $t \in [0, \tau]$ and selects from each jump of P_t the Hawkes jumps of $X_t^{i,k}$ with probability $\frac{\lambda_t^{i,k}}{M}$, or no jump at all with probability $\frac{M-\sum_{i,k} \lambda_t^{i,k}}{M}$.

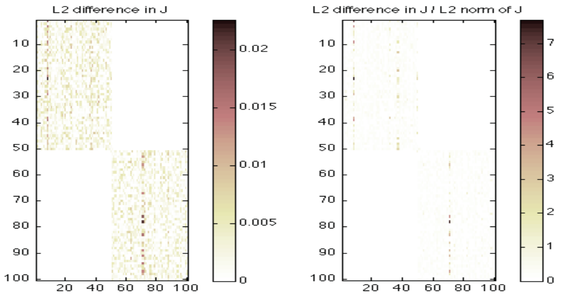

Figure 2: Heatmap of L^2 differences (absolute and relative) between entries of true J and estimated J.

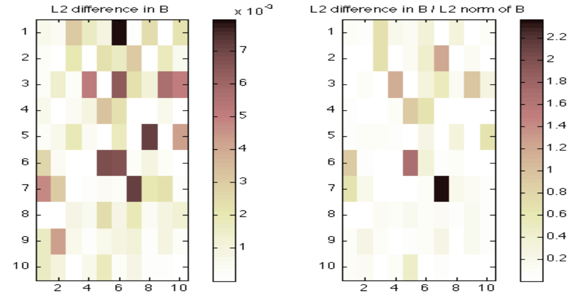

Figure 3: Heatmap of L^2 differences (absolute and relative) between entries of true B and estimated B.

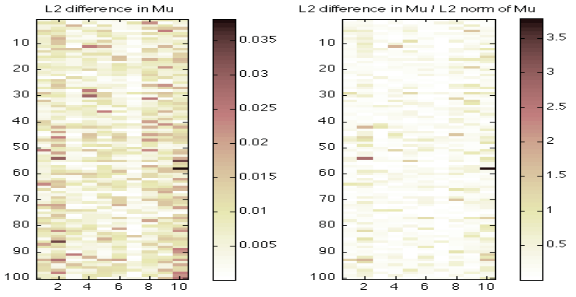

Figure 4: Heatmap of L^2 differences (absolute and relative) between entries of true μ and estimated μ.

[5]) - which guarantees a more realistic view of the information diffusion cascades. Also, the model takes into consideration not only interactions between users (as in [29]) but also interactions between topics (as in [19]), which provides a deeper analysis of influences in social networks.

Another crucial advantage of this framework is that all the parameters are versatile and allow for a variety of extensions and adaptations for real-life situations: if one has predefined labelled data, if one wants to discover the topics broadcasted in the messages, if one wants to change the shape of the temporal kernel in order to exploit a longer

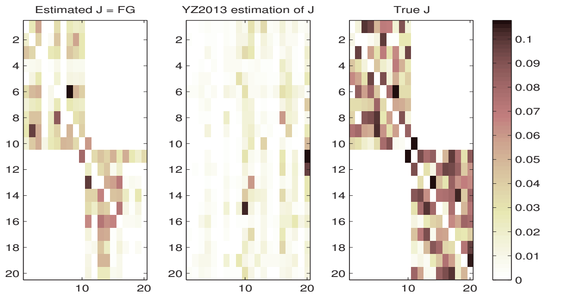

Figure 5: Left: our proposed estimation. Center: estimation following [29]. Right: true J.

range of interactions, if one wants to model information diffusion in more than one social network at the same time, etc...

Nonnegative matrix factorization techniques are interesting here for mainly two reasons: the multiplicative updates derived from the optimization problem are easy to implement, even in a distributed fashion - they are basically matrix products and entrywise operations - and the complexity of the algorithm is linear in the data, allowing one to perform estimations in real-life social networks, especially if some of the parameters are already known beforehand.

One can also notice that by performing a dimensionality reduction $J = FG$ during our nonnegative matrix factorization estimation, we not only estimated the influence that users have on one another but we also acquired information on the communities of the underlying social network, since we were able to factorize the hidden influence graph J. Here, we used heavily the self-exciting model to retrieve the hidden influence graph, which is different from other graphs generated by different methods; for example, one could weight the communication graph A with the number of messages from one user to its neighbours, but by doing so one looses the temporal character. Moreover, the graphs found by performing this kind of technique are under the assumption that messages influence directly other users, which may not be the case. In our Hawkes framework, the influence is a byproduct of the interaction of users and information, and therefore their influence is probabilistic - it may or may not occur at each broadcast.

6. REFERENCES

[1] R. Albright, J. Cox, D. Duling, A. N. Langville, and C. D. Meyer. Algorithms, initializations, and convergence for the nonnegative matrix factorization. *SAS Technical report, ArXiv: 1407.7299*, 2014.

[2] E. Bacry and J.-F. Muzy. Second order statistics characterization of Hawkes processes and non-parametric estimation. *ArXiv: 1401.0903v1*, 2014.

[3] C. Boutsidisa and E. Gallopoulos. Svd based initialization: A head start for nonnegative matrix factorization. *Pattern Recognition*, 41:1350–1362, 2008.

[4] R. Crane and D. Sornette. Robust dynamic classes revealed by measuring the response function of a social system. *Proceedings of the National Academy of*

Sciences, 105(41):15649–15653, 2008.

[5] D. J. Daley and D. Vere-Jones. *An introduction to the theory of point processes*. Springer series in Statistics. Springer, 2005.

[6] C. Févotte and J. Idier. Algorithms for nonnegative matrix factorization with the β-divergence. *Neural Computation*, 23(9):2421–2456, Sep. 2011.

[7] M. Gomez-Rodriguez, D. Balduzzi, and B. Schölkopf. Uncovering the temporal dynamics of diffusion networks. *International Conference on Machine Learning (ICML)*, pages 561–568, 2011.

[8] M. Gomez-Rodriguez and B. Schölkopf. Submodular inference of diffusion networks from multiple trees. *International Conference on Machine Learning (ICML)*, 2012.

[9] P. F. Halpin. An EM algorithm for Hawkes process. *Proceedings of the 77th Annual Meeting of the Psychometric Society*, 2013.

[10] A. G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58:83–90, 1971.

[11] R. Kompass. A generalized divergence measure for nonnegative matrix factorization. *Neural Computation*, 19(3):780–791, 2007.

[12] V. Krishnamurthy and A. d'Aspremont. Convex algorithms for nonnegative matrix factorization. *ArXiv: 1207.0318*, May 2007.

[13] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[14] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. *Advances in Neural and Information Processing Systems (NIPS) 13*, pages 556–562, 2001.

[15] E. Lewis and G. O. Mohler. A nonparametric EM algorithm for multiscale Hawkes processes. *Journal of Nonparametric Statistics*, pages 1–16, 2011.

[16] L. Li and H. Zha. Dyadic event attribution in social networks with mixtures of Hawkes processes. *Proceedings of 22nd ACM International Conference on Information and Knowledge Management (CIKM)*, 2013.

[17] T. Liniger. Multivariate Hawkes processes. *ETH Doctoral Dissertation*, (18403), 2009.

[18] H. Minc. *Nonnegative Matrices*. John Wiley & Sons, New York, NY, USA, 1988.

[19] S. Myers and J. Leskovec. Clash of the contagions: Cooperation and competition in information diffusion. *IEEE International Conference On Data Mining (ICDM)*, 2012.

[20] S. Myers, J. Leskovec, and C. Zhu. Information diffusion and external influence in networks. *KDD '12: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2012.

[21] Y. Ogata. The asymptotic behaviour of maximum likelihood estimators for stationary point processes. *Annals of the Institute of Statistical Mathematics*, 30:Part A:243–261, 1978.

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

[23] S. A. Pasha and V. Solo. Hawkes-laguerre dynamic index models for point processes. *Proceedings of 52nd IEEE Conference on Decision and Control (CDC)*, 2013.

[24] S. A. Pasha and V. Solo. Hawkes-laguerre reduced rank model for point process. *Proceedings of 38th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 6098–6102, 2013.

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

[26] T. M. Snowsill, N. Fyson, T. D. Bie, and N. Cristianini. Refining causality: who copied from whom? *ACM SIGKDD 2011*, pages 466–474, 2011.

[27] V. Y. F. Tan and C. Févotte. Automatic relevance determination in nonnegative matrix factorization with the β-divergence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1592–1605, 2013.

[28] S. Wild, J. Curry, and A. Dougherty. Improving non-negative matrix factorizations through structured initialization. *Pattern Recognition*, 37, 2004.

[29] S.-H. Yang and H. Zha. Mixture of mutually exciting processes for viral diffusion. *Proceedings of the 30th International Conference on Machine Learning (ICML)*, 2013.

[30] R. J. Ypma, A. M. Bataille, A. Stegeman, G. Koch, J. Wallinga, and W. M. van Ballegooijen. Unravelling transmission trees of infectious diseases by combining genetic and epidemiological data. *Procedings of the Royal Society B*, 279:444–450, 2012.

[31] K. Zhou, H. Zha, and L. Song. Learning social infectivity in sparse low-rank networks using multi-dimensional Hawkes processes. *Proceedings of the 16th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2013.

[32] K. Zhou, H. Zha, and L. Song. Learning triggering kernels for multi-dimensional Hawkes processes. *Proceedings of the 30th International Conference on Machine Learning (ICML), 2013*, 2013.