Transfer Hawkes processes with content information

Li, Tianbo; Wei, Pengfei; Ke, Yiping

2018

Li, T., Wei, P., & Ke, Y. (2018). Transfer Hawkes processes with content information. Proceedings of the 2018 IEEE International Conference on Data Mining (ICDM), 1116-1121. doi:10.1109/icdm.2018.00145

https://hdl.handle.net/10356/142999

https://doi.org/10.1109/ICDM.2018.00145

Transfer Hawkes Processes with Content Information

Tianbo Li
Nanyang Technological University
Singapore, 639798
Email: tianbo001@e.ntu.edu.sg

Pengfei Wei
Nanyang Technological University
Singapore, 639798
Email: pwei001@e.ntu.edu.sg

Yiping Ke
Nanyang Technological University
Singapore, 639798
Email: ypke@ntu.edu.sg

***Abstract*—Hawkes processes are widely used for modeling event cascades. However, content and cross-domain information which is also instrumental in modeling is usually neglected. In this paper, we propose a novel model called transfer Hybrid Least Square for Hawkes (trHLSH) that incorporates Hawkes processes with content and cross-domain information. We also present the effective learning algorithm for the model. Evaluation on both synthetic and real-world datasets demonstrates that the proposed model can jointly learn knowledge from temporal, content and cross-domain information, and has better performance in terms of network recovery and prediction.**

***Keywords*-Hawkes processes, transfer learning, event cascades**

I. Introduction

Hawkes processes [1] are widely used to model complicated event sequences produced from natural and social systems. Hawkes processes can capture both individual and interactive behaviors, and have achieved satisfactory results in a variety of disciplines, such as Finance [2], Seismology [3], and Neurophysiology [4]. Hawkes process is a data-driven model for social network analysis, which is competent for dealing with the ubiquitous data regime in social networks called *event cascades*, or *asynchronous event sequences*[1]. A cascade implies a successive process consisting of a series of events. In social network services, events refer to daily online behaviors such as posting, commenting, retweeting or sharing. Event cascades can be epitomized by timelines. Generally, there exists certain triggering pattern among events. Information in social networks regularly disseminates in such a way that one event triggers a series of responses from other users. For example, when a meme goes viral in social networks, users see posts from each other, get intrigued, and comment or share. Hawkes processes have witnessed many successes in handling event cascades leveraging temporal information in them with applications in network structure recovery [5], community detection [6], recommendation [7], etc.

Although Hawkes processes cater to the need for dealing with temporal information from event cascades and have achieved satisfactory results, some content information, which is also conducive to improving accuracy and robustness for various tasks, should also be taken into account. In this case Hawkes processes can be applied to capture the mutual influence. Neglecting content information, Hawkes model may be deceived. Therefore a more accurate model that can effectively incorporate content information is on demand.

Besides content information, cross-domain[2] knowledge can also be beneficial. For example in the cold-start problem, suppose we would like to build recommender systems for the e-commerce start-ups, but we do not have adequate data on hand. In this case, we can use data from mature businesses like Amazon and transfer the knowledge across platforms. Similar application scenarios are commonly encountered. Recently, transfer learning [8], which has been a topic of active interest, sheds light on how to exploit the knowledge from other domains and help improve the performance of models. However to the best of our knowledge, none of the existing works have explored transfer learning for Hawkes processes.

In this paper, we investigate the idea of augmenting Hawkes processes with both content and cross-domain information, and we propose a novel model called transfer **H**ybrid **L**east **S**quare for **H**awkes (trHLSH). The trHLSH is based on the least square estimation of Hawkes and auto-regression of feature vectors. A regularizer term is added to control that estimation of parameters on the target domain will not deviate too much from those on the source domain. The model captures all of the three aforementioned types of information:

- temporal information of event cascades;
- event content information;
- cross-domain knowledge.

The noteworthy novelties and contributions of this paper can be summarized as follows:

- To the best of our knowledge, the model we propose is the first attempt to deal with temporal and content information, as well as cross-domain transfer simultaneously.

[1]In this paper, these two terms are interchangeable.

[2]This is a concept that is widely used in the studies of transfer learning. We give the definition in Section III-D.

- We test our model on the data crawled from Facebook and Twitter. The experimental results on both synthetic and real-world data demonstrate the superiority of our model in terms of parameter recovery and prediction.

The rest of the paper is organized as follows. Section II gives a general overview of related works. Section III presents some fundamental knowledge and defines the notations that are used later. Section IV discusses our proposed model and its learning algorithm. Section V reports the experimental results. Finally, Section VI concludes the paper.

II. Related Works

Hawkes processes have been attracting increasing attention in academia recently. These models generally focus on recovering the hidden network of social influence, which is represented by the infectivity parameter α's, from the observable asynchronous event sequences, or so-called *Granger causality*. [9] proposes a convex optimization approach to discover the hidden network based on regularized Hawkes processes which can capture low-rank and sparse structure on network topology. [10] replaces the decay kernels $\kappa(t)$ with a sparse log Gaussian Cox process in order to mimic the mutual influences. They also present a fully-Bayesian, parallel inference algorithm by using the Poisson superposition principle. [11] introduces an effective method using a series of decay kernels instead of one and recovers the Granger causality graph via group sparsity. However, almost all of these models regarding Hawkes processes just utilize the temporal information and the corresponding content information is neglected.

Besides, there are some explorations in the context of natural language processing. [12] addresses diffusion network inference and meme tracking via viral Twitter texts. The proposed model considers content information for network recovery. Dirichlet-Hawkes Processes proposed in [13] takes into account both textual contents and temporal information with application to clustering document streams. The model can recover both topics and temporal dynamics. [14] proposes a clustering method for asynchronous event sequences using a Dirichlet mixture model. None of the above works, however, discussed incorporating cross-domain information. Explorations on combining Hawkes processes and transfer learning are lacking. To the best of our knowledge, this paper is the first attempt along this line.

III. Preliminaries

A. Notations

To facilitate the subsequent discussion, frequently used symbols and their definitions are listed in Table I. Note that we refer to the nodes in social networks or any entities that generate events as **users**. Each user is represented as a dimension in multi-dimensional Hawkes processes.

Table I
Table of Notations

Symbol	Description
$t_k^i \in [0, T]$	Time of the k-th event occurring in the i-th dimension, during the observation window $[0, T]$.
$\boldsymbol{f}_k^i \in \mathbb{R}^d$	d-dimensional event feature corresponding to t_k^i.
$n^i \in \mathbb{N}_+$, $n = \sum_{i=1} n^i$	Total number of events that occur in the i-th dimension. n is the sum over all dimensions.
M	Total number of dimensions or users.
$\boldsymbol{N}(t) = \{N^i(t)\}_{i=1}^M$	M-dimensional counting precess, or specifically refers to Hawkes process.
$\lambda^i(t)$	Conditional intensity function of user i. Without otherwise stated, the intensity function is conditioned on history $\mathcal{H}_-$.
$\tau^i(\boldsymbol{f}\|t)$	Conditional probability density function of feature vector given time t.
$\mu^i \in \mathbb{R}_+$	Base intensity of the i-th dimension which controls the probability of immigrants arrival.
$\alpha^{ij} \in \mathbb{R}_+ \cup \{0\}$	Infectivity parameter that determines the influence on the i-th user from the j-th user.
$\kappa(t\|\beta^{ij})$	Kernel function with parameter $\beta^{ij} \in \mathbb{R}_+$ which parameterizes decay of infection w.r.t. time.
$\boldsymbol{S}_T, \boldsymbol{S}_S$	Event sequences (including timestamps and features) of target and source domain, respectively.

B. Hawkes Processes

Hawkes processes [1], a class of point processes, are essentially multi-dimensional nonhomogeneous Poisson processes, which traditionally play a central role in modeling the self- and mutually exciting behavior of events. A Hawkes process $N(t)$ is characterized by its conditional intensity function $\lambda(t|\mathcal{H}_{t_-})$. A M-dimensional Hawkes process $\boldsymbol{N}(t)$, $t \in [0, T]$ is given by the intensities $\lambda^i(t), i = 1, \ldots, M$, as follows,

$$\lambda^i(t) = \mu^i + \sum_{j=1}^{M} \sum_k \alpha^{ij} \kappa(t - t_k^j | \beta^{ij}). \tag{1}$$

Here and hereafter we use $\lambda(t)$ instead of $\lambda(t|\mathcal{H}_{t_-})$ for abbreviation unless otherwise stated. $\mu^i \in \mathbb{R}_+$ is the base intensity of the i-th dimension. $\alpha^{ij} \in \mathbb{R}_+$ is called infectivity pamameter. It measures the influence on the i-th user from the j-th user. M is the dimensionality. $t_k^j \in [0, T]$ is the timestamp recording the k-th event of those happening in the j-th dimension. $\kappa(t)$ is the decay kernel function with bandwidth β^{ij}. It is worth noting that the overwhelming majority of existing works assume μ^i's and α^{ij}'s are positive as we do, which means only excitation is considered. Recently some study such as [15] allows inhibition through negative valued α^{ij}'s. Choices of kernel functions include but are not limited to exponential kernel, Gaussian kernel, power-law kernel and Rayleigh kernel. In this study, we use

exponential kernel throughout.

C. Least Square Estimator

The least square estimator for nonhomogeneous Poisson process is a special case of a wide class of estimators, namely M-estimators as explained in [16]. The objective is to minimize the product of conditional intensity function and its deviation from the actual sample of the process $\widetilde{N}^i$:

$$\min \int_0^T \lambda^i(t) \left(\lambda^i(t)dt - d\widetilde{N}^i(t) \right). \tag{2}$$

Note that maximum likelihood estimator also belongs to the class of M-estimators. However in the case of Hawkes process, maximum likelihood estimator is more computationally difficult than least square estimator. It cannot result in a "one-step" solution as least square estimator does. Least square estimator can also give satisfactory final estimates from the same asymptotic distribution as maximum likelihood estimators do [16]. Therefore we adopt least square estimator as the starting point of our model trHLSH, which is discussed later.

For convenience we introduce notations:

$$\boldsymbol{\theta}_i = \left(\mu^i, \alpha^{i1}, \alpha^{i2}, \ldots, \alpha^{iM}\right)', \tag{3}$$

$$\boldsymbol{x}_i(t) = \left(1, \sum_k \kappa(t - t_k^1|\beta), \ldots, \sum_{k'} \kappa(t - t_{k'}^M|\beta)\right)'. \tag{4}$$

The intensity function $\lambda^i(t)$ as shown in Eq.(1) can be recasted as the sum of: $\boldsymbol{\theta}_i$, the parameter vector to be inferred, and $\boldsymbol{x}_i(t)$, the kernel sum vector. Applying the notations, Eq.(2) is equivalent to:

$$\min_{\boldsymbol{\theta}_i} \int_0^T \boldsymbol{\theta}_i' \boldsymbol{x}_i(t) \boldsymbol{x}_i(t)' \boldsymbol{\theta}_i dt - 2 \int_0^T \boldsymbol{\theta}_i' \boldsymbol{x}_i(t) d\widetilde{N}^i(t). \tag{5}$$

Let

$$\boldsymbol{Z}_i = \int_0^T \boldsymbol{x}_i(t) \boldsymbol{x}_i(t)' dt, \tag{6}$$

and

$$\boldsymbol{y}_i = \int_0^T \boldsymbol{x}_i(t) d\widetilde{N}^i(t) = \sum_{t_k^i} \boldsymbol{x}_i(t_k^i). \tag{7}$$

The objective function becomes,

$$\begin{aligned} \min_{\boldsymbol{\theta}} \quad & \boldsymbol{\theta}' \boldsymbol{Z} \boldsymbol{\theta} - 2\boldsymbol{\theta}' \boldsymbol{y} \\ \text{s.t.} \quad & \boldsymbol{\theta} \geqslant \mathbf{0} \end{aligned} \tag{8}$$

Here the subscript denoting the dimension i is omitted. This optimization problem can be directly solved by quadratic programming.

D. Transfer Learning

Transfer learning [8] aims to transfer knowledge from one domain (called the *source domain*) to help the tasks of another domain (called the *target domain*). Typically, the source and target domains are related in the sense that they share some common knowledge which can be transferred across domains. Based on the knowledge transferred, transfer learning is categorized into the feature-based one, the instance-based one, and the parameter-based one. In this paper, we focus on the parameter transfer, which is to discover the relationship of parameters, between two different Hawkes processes. Specifically, in the experiments on real-world data, we use the event sequences from Twitter as source domain and those from Facebook as target domain. We assume that the network structures of both domains are similar and transferable. Therefore in the proposed model, the knowledge we transfer is actually the infectivity parameter α's, which describe the network structure.

IV. The Proposed Model

In this section we present the proposed model and its learning algorithm.

A. Leveraging Content Information

In Section III-C we describe the least square estimator for Hawkes process that models temporal information. Now we illustrate an auto-regression model for incorporating the content information. For an event which occurs at t, its content can be represented as a feature vector $\boldsymbol{f}(t)$, or $\boldsymbol{f}_k^i$ corresponding to t_k^i, via feature embedding. Assume content and time are independent, and $\boldsymbol{f}^i(t)$ can be defined by all the events (with their contents) that happen prior to t as,

$$\boldsymbol{f}^i(t) = \sum_{j=1}^{M} \frac{1}{N^j(t)} \sum_{k=1}^{N^j(t)} w_k^{ij} \boldsymbol{f}_{N^j(t)+1-k}^j + w_0^i \mathbf{1} + \epsilon, \tag{9}$$

where w_k^{ij} is the regression coefficient, w_0^i the intercept and ϵ the Gaussian noise. $\mathbf{1}$ is a vector with each entry one. The intuition behind is what you may post depends on what you have read. The content is modeled as a linear combination of the historical events. However the number of coefficients equals to the total number of events, which makes the estimation intractable as a result of the short rank of the design matrix. Therefore we assume that all the features of the same dimension share the same coefficient. Under this assumption, Eq.(9) becomes,

$$\boldsymbol{f}^i(t) = \sum_{j=1}^{M} \frac{w^{ij}}{N^j(t)} \sum_{k=1}^{N^j(t)} \boldsymbol{f}_k^j + w_0^i \mathbf{1} + \epsilon, \tag{10}$$

The least square estimation for w^{ij} is,

$$\min_{w^{ij}} \sum_{i=1}^{M} \sum_{k=1}^{n^i} \left\| \boldsymbol{f}_k^i - \sum_{j=1}^{M} \frac{1}{N^j(t_k^i)} \sum_{k'=1}^{N^j(t_k^i)} w^{ij} \boldsymbol{f}_{k'}^j - w_0^i \mathbf{1} \right\|_2^2. \tag{11}$$

Let

$$\boldsymbol{F}_k^i = \left(\mathbf{1}, \frac{1}{N^1(t_k^i)} \sum_{k'=1}^{N^1(t_k^i)} \boldsymbol{f}_{k'}^j, \ldots, \frac{1}{N^M(t)} \sum_{k'=1}^{N^M(t_k^i)} \boldsymbol{f}_{k'}^j \right), \quad (12)$$

$$\boldsymbol{w}^i = \left(w_0^i, w^{i1}, \ldots, w^{iM} \right)', \quad (13)$$

$$\boldsymbol{\Psi}^i = \sum_{k=1}^{n^i} \left(\boldsymbol{F}_k^i \right)' \boldsymbol{F}_k^i, \quad (14)$$

$$\boldsymbol{\phi}^i = \sum_{k=1}^{n^i} \left(\boldsymbol{F}_k^i \right)' \boldsymbol{f}_k^i. \quad (15)$$

Eq.(11) can be rewritten as an optimization problem,

$$\min_{\boldsymbol{w}} \quad \boldsymbol{w}' \boldsymbol{\Psi} \boldsymbol{w} - 2 \boldsymbol{w}' \boldsymbol{\phi}. \quad (16)$$

Here the superscript i is also omitted.

B. Hybrid Least Square for Hawkes (HLSH)

Note that α^{ij} quantifies the influence from dimension j to i which is inferred by temporal information. Symmetrically w^{ij} also reflects the influence from dimension j to i which is, however, inferred by the content information. Since α^{ij} and w^{ij} both reflect the network structure, they should be similar. Therefore we impose a $L2$ norm regularizer of $\boldsymbol{\theta} - \boldsymbol{w}$. Combining Eq.(8) and Eq.(16), now we have the objective function of HLSH:

$$\begin{aligned} \min_{\boldsymbol{\theta}, \boldsymbol{w}} \quad & \boldsymbol{\theta}' \boldsymbol{Z} \boldsymbol{\theta} - 2 \boldsymbol{\theta}' \boldsymbol{y} + \eta_1 \left(\boldsymbol{w}' \boldsymbol{\Psi} \boldsymbol{w} - 2 \boldsymbol{w}' \boldsymbol{\phi} \right) + \eta_2 \left\| \boldsymbol{\theta} - \boldsymbol{w} \right\|_2^2, \\ \text{s.t.} \quad & \boldsymbol{\theta} \geqslant \mathbf{0}. \end{aligned} \quad (17)$$

where η_1 and η_2 are hyperparameters of the regularizations.

C. Transfer HLSH

We now take into account cross-domain knowledge for augmenting the HLSH model. The objective function is formulated as,

$$\begin{aligned} \min_{\boldsymbol{\theta}_T, \boldsymbol{w}_T} \quad & \boldsymbol{\theta}_T' \boldsymbol{Z} \boldsymbol{\theta}_T - 2 \boldsymbol{\theta}_T' \boldsymbol{y} + \eta_1 \left(\boldsymbol{w}_T' \boldsymbol{\Psi} \boldsymbol{w}_T - 2 \boldsymbol{w}_T' \boldsymbol{\phi} \right) + \\ & \eta_2 \left\| \boldsymbol{\theta}_T - \boldsymbol{w}_T \right\|_2^2 + \eta_3 \left\| \boldsymbol{\theta}_T - \boldsymbol{\theta}_S \right\|_2^2 + \eta_4 \left\| \boldsymbol{w}_T - \boldsymbol{w}_S \right\|_2^2, \\ \text{s.t.} \quad & \boldsymbol{\theta}_T \geqslant \mathbf{0}. \end{aligned} \quad (18)$$

Here $\boldsymbol{\theta}_S$ and $\boldsymbol{w}_S$ are the parameters that are pre-learned from the source domain by HLSH. The last three regularization terms and their meanings are:

- $\left\| \boldsymbol{\theta}_T - \boldsymbol{w}_T \right\|_2^2$: constraint on similarity of the network structure learned from temporal and content information;
- $\left\| \boldsymbol{\theta}_T - \boldsymbol{\theta}_S \right\|_2^2$ and $\left\| \boldsymbol{w}_T - \boldsymbol{w}_S \right\|_2^2$: constraints on similarity of the parameters learned from the target and source domains.

Algorithm 1 presents the procedure of the learning algorithm above.

Algorithm 1: trHLSH: Transfer Hybrid Least Square for Hawkes

input : $\boldsymbol{S}_S, \boldsymbol{S}_T, \eta_1, \eta_1', \eta_2, \eta_2', \eta_3, \eta_4, \kappa(\cdot)$
output: $\boldsymbol{\theta}_S, \boldsymbol{\theta}_T, \boldsymbol{w}_S, \boldsymbol{w}_T$
Calculate $\boldsymbol{Z}_S, \boldsymbol{y}_S, \boldsymbol{\Psi}_S, \boldsymbol{\phi}_S, \boldsymbol{Z}_T, \boldsymbol{y}_T, \boldsymbol{\Psi}_T, \boldsymbol{\phi}_T$ by Eq.(7), (6), (14), (15);
Calculate $\boldsymbol{\theta}_S$ and $\boldsymbol{w}_S$ using η_1', η_2' and Eq.(17);
Calculate $\boldsymbol{\theta}_T$ and $\boldsymbol{w}_T$ using η_1, η_2, η_3, η_4 and Eq.(18);
return $\boldsymbol{\theta}_S$, $\boldsymbol{\theta}_T$, $\boldsymbol{w}_S$, $\boldsymbol{w}_T$.

V. Experiments

A. Synthetic Data

We first test our model on synthetic data in terms of the performance of parameter recovery. Then we test on real-world data in terms of prediction.

We follow the method for generating synthetic data used in [10]. First, we generate a 10-dimensional Erdös-Rényi graph with `sparsity` parameter ρ as the adjacency matrix for target domain. If there is an edge, then we generate a weight from a uniform distribution. The weighted adjacency matrix is the α's (infectivity matrix) we use in Hawkes process. Before generating event cascades, the stability condition is checked [17] such that the simulation does not lead to infinite numbers of events. Then we randomly add or remove edges in the Erdös-Rényi graph that we have generated at the first step with a small probability as the adjacency matrix for source domain and we generate a new weight for each newly-added edge. We use `similarity` to describe how many edges are the same in both adjacency matrices of source and target domains. Next we generate base intensities μ's uniformly. We adopt exponential kernels and set $\beta = 1$. After all the parameters needed are prepared, we apply the thinning algorithm [18] and branching structure [19] to simulate two event cascades for source and target domains respectively. `Time ratio` of observation windows measures how much more information is in source domain than in target domain. We fix the observation window of target domain as 60, and change that of source domain accordingly. Note that in the brunching structure, the parent event can be indicated. When an event triggers an offspring, the corresponding feature vector is generated from Gaussian distribution with the parent feature vector as the mean. `Feature bandwidth` is the variance parameter for the Gaussian distribution, which controls how alike the feature vectors are between generations. For each experiment, we generate 50 different samples and apply the proposed models to obtain the estimation of infectivity parameter α's. The performance results reported are the average over all the 50 runs.

The evaluation metrics we use in parameter recovery are:

- `RMSE`: the root-mean-square error of α's.

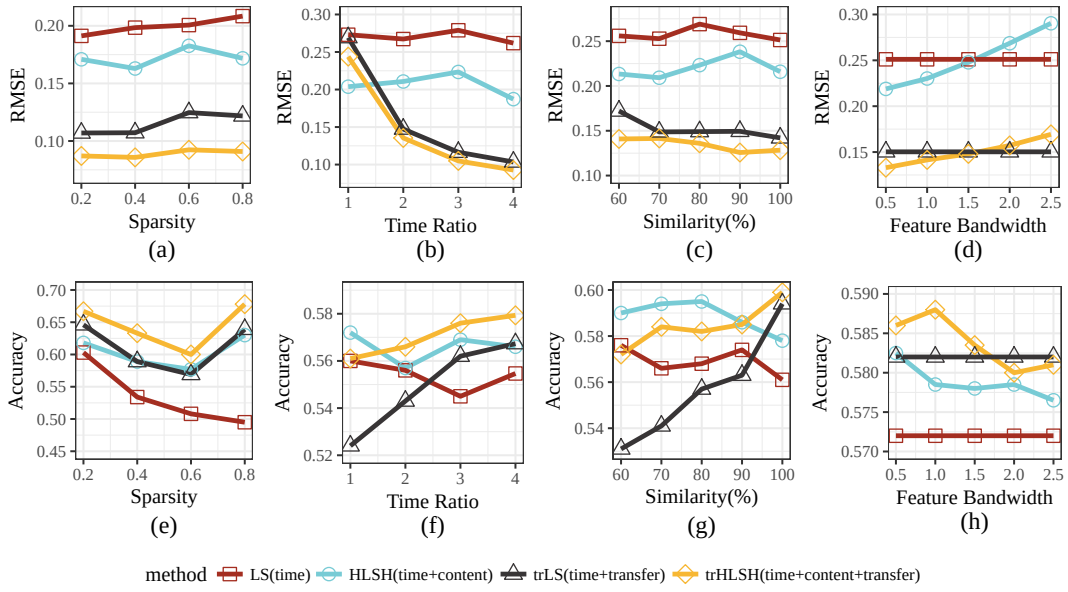


Figure 1. Performance on synthetic data.

- `Accuracy`: the percentage of edges that are correctly predicted by the estimated adjacency matrix. It measures the accuracy of the prediction of adjacency matrices. The estimated adjacency matrix is obtained by bisecting the infectivity matrix with a designated threshold. That is, if α is larger than the threshold, we regard it as an edge.

A lower `RMSE` and a higher `accuracy` indicate better performance for parameter recovery.

We involve 4 models for comparison:

- `LS(time)`. The original Hawkes process. Only temporal information is used.
- `HLSH(time+content)`. This method utilizes both temporal and content information.
- `trLS(time+transfer)`. The `trLS` algorithm is to set $\eta_1 = 0, \eta_2 = 0, \eta_4 = 0$ in Algorithm 1. The method utilizes both temporal and cross-domain information, but not content information.
- `trHLSH(time+content+transfer)`. This method is shown in Algorithm 1, which leverages all of temporal, cross-domain and content information.

Fig. 1 shows the performance of the 4 models when varying `sparsity`, `similarity`, `time ratio` and `feature bandwidth`. Fig.1(a) and 1(e) show how sparsity affects performance. Fig.1(b) and 1(f) demonstrate that the more informative source domain is, the more helpful transfer will be. Fig.1(c) and 1(g) illustrate how the similarity of the networks of source and target domains affects cross-domain knowledge transfer. The more alike two domains are, such information will be more useful for transfer. Fig.1(d) and 1(h) show the influence of content information. The more alike between generations (a smaller feature bandwidth), the more helpful content information will be. In general, `trHLSH` outperforms `trLS`, `HLSH`, whereas `LS` is less preferable to the others.

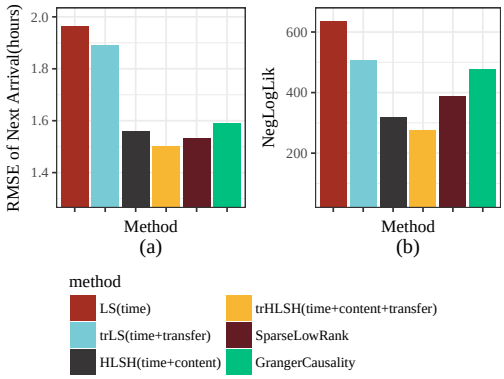
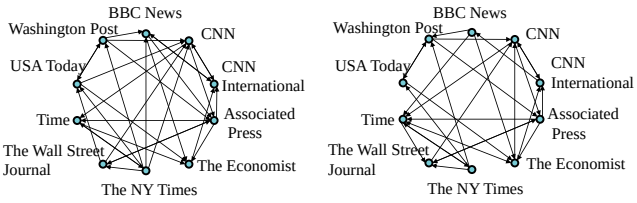


Figure 2. Performance on real-world data.

B. *Real-world Data*

We also evaluate our proposed model in real-world data crawled from Facebook and Twitter. We crawled 1482 posts from Facebook as the target domain and 1587 posts from Twitter as the source domain of 10 same major news agencies including CNN, BBC, Associated Press, the New York Times , the Wall Street Journal, Washington Post, etc. We divide Facebook dataset into 70% and 30% as training and testing datasets, respectively.

The textual contents are represented by bag-of-words. We extract 2000 most frequent words as features. Then



(a) Facebook (b) Twitter

Figure 3. Network Structure learned from Facebook and Twitter, respectively.

we apply a simple principal component analysis (PCA) for dimensionality reduction. Eventually we obtain the features with the first 200 principle components.

After that, we train our model and the baseline models on the training dataset. In addition to the three baseline models tested on synthetic data, we include two more baselines in this experiment:

- `SparseLowRank` [9]. A nuclear and ℓ_1 norm of infectivity matrix is imposed to the general Hawkes process, which takes into account the prior knowledge of sparse and low-rank structure in social networks.
- `GrangerCausality` [11]. This model replaces the commonly used exponential decay kernels with a series of Gaussian basis functions, which can flexibly capture the mutual influences.

We evaluate the trained models on the test dataset, in terms of the `RMSE` of predicting the next arrival time and the negative log likelihood `NegLogLik` of the test dataset. The results are shown in Fig. 2. It can be seen that `trHLSH` outperforms the baselines in both metrics. It has the lowest `RMSE` on predicting next arrival time, and also the lowest `NegLogLik`. This result validates that content and cross-domain information are instrumental in improving predictive performance of Hawkes processes. The network structures learned from Facebook and Twitter are presented in Fig. 3. Some similarities can be seen between the two graphs.

VI. Conclusions

In this paper, we present a novel model that organically leverages temporal, content and cross-domain information. The proposed model augments the basic Hawkes process by taking into account features associated with events and transferring the network structure inferred from a source domain. The trHLSH can be learned efficiently by quadratic programming, which enjoys many computational merits. The experiential results on both synthetic data and real-world data crawled from Facebook and Twitter suggest that our model has a better performance than the baseline models in terms of network structure recovery and prediction.

Acknowledgement

This research is supported in part by the AcRF Tier-1 Grant (RG135/14) from Ministry of Education of Singapore. The authors would like to thank Mr. Yuchong Zhang for data preparation.

References

[1] A. G. Hawkes, "Point spectra of some mutually exciting point processes," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 438–443, 1971.

[2] C. G. Bowsher, "Modelling security market events in continuous time: Intensity based, multivariate point process models," *Journal of Econometrics*, vol. 141, no. 2, pp. 876–912, 2007.

[3] Y. Ogata, "Statistical models for earthquake occurrences and residual analysis for point processes," *Journal of the American Statistical association*, vol. 83, no. 401, pp. 9–27, 1988.

[4] D. R. Brillinger, H. L. Bryant, and J. P. Segundo, "Identification of synaptic interactions," *Biological cybernetics*, vol. 22, no. 4, pp. 213–228, 1976.

[5] K. Zhou, H. Zha, and L. Song, "Learning social infectivity in sparse low-rank networks using multi-dimensional hawkes processes," in *AISTATS*, 2013, pp. 641–649.

[6] C. Blundell, J. Beck, and K. A. Heller, "Modelling reciprocating relationships with hawkes processes," in *NIPS*, 2012, pp. 2600–2608.

[7] S. A. Hosseini, K. Alizadeh, A. Khodadadi, A. Arabzadeh, M. Farajtabar, H. Zha, and H. R. Rabiee, "Recurrent poisson factorization for temporal recommendation," in *KDD '17*. New York, NY, USA: ACM, 2017, pp. 847–855.

[8] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2010.

[9] K. Zhou, H. Zha, and L. Song, "Learning triggering kernels for multi-dimensional hawkes processes," in *ICML'13*, 2013, pp. 1301–1309.

[10] S. Linderman and R. Adams, "Discovering latent network structure in point process data," in *ICML*, 2014, pp. 1413–1421.

[11] H. Xu, M. Farajtabar, and H. Zha, "Learning granger causality for hawkes processes," in *ICML*, 2016, pp. 1717–1726.

[12] S.-H. Yang and H. Zha, "Mixture of mutually exciting processes for viral diffusion," in *ICML*, 2013, pp. 1–9.

[13] N. Du, M. Farajtabar, A. Ahmed, A. J. Smola, and L. Song, "Dirichlet-hawkes processes with applications to clustering continuous-time document streams," in *SIGKDD'15*. ACM, 2015, pp. 219–228.

[14] H. Xu and H. Zha, "A dirichlet mixture model of hawkes processes for event sequence clustering." NIPS, 2017.

[15] P. Reynaud-Bouret, S. Schbath *et al.*, "Adaptive estimation for hawkes processes; application to genome analysis," *The Annals of Statistics*, vol. 38, no. 5, pp. 2781–2822, 2010.

[16] P. K. Andersen, O. Borgan, R. D. Gill, and N. Keiding, *Statistical models based on counting processes*. Springer Science & Business Media, 2012.

[17] D. J. Daley and D. Vere-Jones, *An introduction to the theory of point processes: volume II: general theory and structure*. Springer Science & Business Media, 2007.

[18] Y. Ogata, "On lewis' simulation method for point processes," *IEEE Transactions on Information Theory*, vol. 27, no. 1, pp. 23–31, 1981.

[19] A. G. Hawkes and D. Oakes, "A cluster process representation of a self-exciting process," *Journal of Applied Probability*, vol. 11, no. 3, pp. 493–503, 1974.