Inferring the Hidden Structure of Information Propagation Using Probabilistic Model

Haiwei Ma
Shanghai Jiao Tong University
Shanghai, China 200240
Email: memorydavid@126.com

***Abstract*—When the information spreads in the network, it is easy to observe the happening time of each event while relatively hard to know the whole structure of the network. But knowledge about the edges in the network is of great importance since it enables us to predict, maximize or minimize the influence of certain information, so we need to infer the hidden structure, that is, the edges between each node, from the given knowledge about information propagation flow. It is natural to consider a probabilistic model, which is first published by Manuel Gomez-Rodriguez, JURE LESKOVEC, David Balduzzi and BERNHARD SCH OLKOPF. However, this model ignores external influences outside the network on the information propagation and does not consider the content difference of different information and slow dynamic changes in the network while I make several modifications on it to apply to these situations. Further, their original model sets a window size *T* to sample the cascades to increase the accuracy but it increases large complexity while I move off this window size to reduce the model's complexity and keep simplicity because if some nodes keep uninfected when the information diffusion is going to stop, they are really unlikely to be infected after the observing time *T*, that is, an ended information diffusion process is not likely to revive and become popular again. Unfortunately, because of the heavy work to program and debug all by myself and limit of other resources and time, I do not make experiments on the modified model.**

I. Introduction

In the diffusion process, we can often observe when n-odes(people, tweets, etc.) get infected by a virus, mention a piece of information, buy a product, adopt a new behavior, or, more generally, adopt a *contagion* while the hidden structure of the diffusion network remains unclear. For instance, doctors can know about when a person becomes ill, but they cannot tell who infected the patient or how many exposures were necessary for the infection to take hold; we can find when friends post a tweet or retweet, but we cannot know what is their information resource if they do not write about it; marketers can track when customers buy products or subscribe to services, but cannot observe who influenced customers decisions, how long they took to make up their minds, or when they would pass recommendations on to other customers. In all these scenarios, we observe *where* and *when* but not *how* or *why* information (or a virus, a tweet, a decision) propagates through a network of population. In order to describe *how* and *why* information propagates, the knowledge about relationships between people, that is, the edges in the network, is quite important. So given the times when nodes adopt a set of contagions, the goal would then be to infer the structure of the underlying network. But the external influence, the difference of contents, the dynamic changes in the network should also be considered. For example, a sick patient might not be infected by others but ate some contaminated food. Then his illness is caused by external influence outside his friend circles. When the information is about sports, a person may talk about it with those friends who like sports while when the information is about politics, he may talk about it with a different group of people who are interested in politics. Besides, the relationships between people vary along with time. Some relationships may decay and even vanish while new relationships form and get enhanced. To take these factors into consideration, I revise the original model and then we can analyze the information pathways of real-world events, topics, or content.

II. Proposed Model

A. Problem Statement

We use a directed graph $G = (V, E)$ to model the network. Each node in V represents a user and each edge has a weight $w_{i,j}$ to represent the strength of the relationship between node i and node j and describes how frequently information spreads from node i to node j. If the weight w_{ij} is large, it means that the relationship between user i and user j is close and the information is more likely to propagate between them. If the weight is zero, then user i and user j do not have any relationships. Then G is a actually cluster with each pair of nodes having two directed weighted edges.

Because the behavior of information diffusion is like that of contagion infection, we use the word *contagion* to interchange with the word *information* and use the word *infect* to mean the *diffusion process*. A node getting infected is the same meaning as information spreading to that node.

As the information spreads from infected nodes to uninfected nodes, it creates a *cascade*(Fig.1) represented by an N-dimensional vector $t^c = (t_1^c, ..., t_N^c)$, recording when each of N nodes gets infected by the information, where N is the number of infected nodes before our observation time T. For those uninfected nodes, they do not appear in this vector. In an information propagation setting, each cascade corresponds to a different piece of information and the infection time of a node is simply the time when the node first heard of or mentioned the piece of information. We add another node x to V and all the edges from and to node x to represent the external source

outside the social network. t_x is the time the information first appears in the mass media. Now we have a new graph $G' = (V', E')$ where $V' = V \cup x$ and $E' = E \cup edge(x)$.

Now we have the mathematical interpretation of networks and information diffusion. Given a set of cascades of many different contagions, our goal is to infer the underlying network over which contagions propagated. Importantly, the time-stamps assigned to nodes in each cascade induce a directed acyclic graph (DAG) involving those nodes, which need not to be acyclic in the containing network topology. Thus, it is meaningful to refer to parents and children within a cascade, but not on the network. The DAG structure dramatically simplifies the computational complexity of the inference problem.

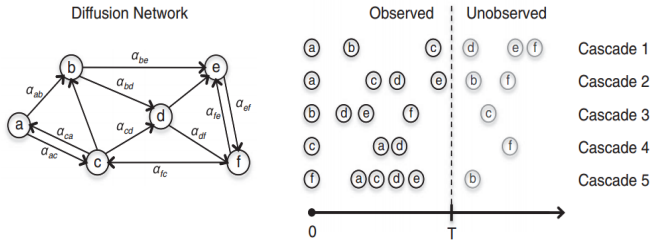

Fig. 1. We observe a set of cascades (right) within an unknown diffusion network (left). For each cascade c, we only observe the times in which nodes get infected up to time T, but not who infected whom. Our goal is to infer the network and transmission rates $\alpha_{i,j}$ based on the observed cascades.

Fig. 1.

B. Probabilistic Model

Because from only observations of a set of cascades of different information, we cannot get the network structure for sure. A probabilistic model is a natural choice to solve the problem. The weight $w_{i,j}$ can also be regarded as the our confidence about the relationships between node i and node j. The more cascades in which we can observe node j infected following node i infected, the more confident we are about the existence and strength of directed edge between node i and j, so the weight $w_{i,j}$ is larger. Clearly this is a generative model in machine learning and we want to get the log likelihood of joint probability of each node infected at their corresponding time recorded in a set of cascades under the condition of our inferred hidden structure. Then we maximize this likelihood to find the best matched hidden network structure, that is, the best parameters $w_{i,j}$. So after we form a probabilistic model, we can use maximum likelihood estimation(MLE) to solve this convex problem.

First we define $f_{in}(\Delta t_{i,j}; w_{i,j})$ as the likelihood of node i infecting node j $\Delta t_{i,j}$ time after node i was infected where $\Delta t_{i,j} = t_j - t_i$ and the parameter $w_{i,j}$ controls the transmission rate. Similarly we define $f_{ex}(\Delta t_{x,j}; w_{x,j})$ as the likelihood of node j getting infected by the external source x $\Delta t_{x,j}$ time after the information first appears at mass media where $\Delta t_{x,j} = t_j - t_x$ and the parameter $w_{x,j}$ controls the transmission rate from the external source x to the internal node j. Note that node j cannot be infected by node i if node i is infected after node j because the infection event obeys the rule of causality. We have defined the probability density function f_{in} and f_{ex} and then comes the corresponding probability cumulative function $F_{in}(\Delta t_{i,j}; w_{i,j})$ and $F_{ex}(\Delta t_{x,j}; w_{x,j})$. Next we define the corresponding survival function as $S_{in}(\Delta t_{i,j}; w_{i,j}) = 1 - F_{in}(\Delta t_{i,j}; w_{i,j})$ and $S_{ex}(\Delta t_{x,j}; w_{x,j}) = 1 - F_{ex}(\Delta t_{x,j}; w_{x,j})$. Then, we define the corresponding hazard function $H_{in}(\Delta t_{i,j}; w_{i,j}) = \frac{f_{in}(\Delta t_{i,j}; w_{i,j})}{S_{in}(\Delta t_{i,j}; w_{i,j})} = -\frac{S'_{in}(\Delta t_{i,j}; w_{i,j})}{S_{in}(\Delta t_{i,j}; w_{i,j})}$ and $H_{ex}(\Delta t_{x,j}; w_{x,j}) = \frac{f_{ex}(\Delta t_{x,j}; w_{x,j})}{S_{ex}(\Delta t_{x,j}; w_{x,j})} = -\frac{S'_{ex}(\Delta t_{x,j}; w_{x,j})}{S_{ex}(\Delta t_{x,j}; w_{x,j})}$. The hazard function means instantaneous infection rate, that is, the infection rate after time $\Delta t_{i,j}$ or $\Delta t_{x,j}$ conditioned on survival for time $\Delta t_{i,j}$ or $\Delta t_{x,j}$. Finally, to get the joint probability, we need to compute the single probability of one node j infected at moment t_j in one cascade c under the condition of the hidden structure. We define $f_j(t_j)$ as the probability density of node j getting infected at moment t_j and we calculate it by the following equations:

$$
\begin{aligned}
f_j(t_j; W) &= f_{ex}(\Delta t_{x,j}; w_{x,j}) \prod_{k \in c, t_k < t_j} S_{in}(\Delta t_{k,j}; w_{k,j}) + \\
&\sum_{i \in c, t_i < t_j} f_{in}(\Delta t_{i,j}; w_{i,j}) \prod_{k \in c, k \neq i, t_k < t_j} S_{in}(\Delta t_{k,j}; w_{k,j}) S_{ex}(\Delta t_{x,j}; w_{x,j}) \\
&= \prod_{k \in c, t_k < t_j} S_{in}(\Delta t_{k,j}; w_{k,j}) S_{ex}(\Delta t_{x,j}; w_{x,j}) \left(\frac{f_{ex}(\Delta t_{x,j}; w_{x,j})}{S_{ex}(\Delta t_{x,j}; w_{x,j})} + \right. \\
&\left. \sum_{i \in c, t_i < t_j} \frac{f_{in}(\Delta t_{i,j}; w_{i,j})}{S_{in}(\Delta t_{i,j}; w_{i,j})} \right) \\
&= \prod_{k \in c, t_k < t_j} S_{in}(\Delta t_{k,j}; w_{k,j}) S_{ex}(\Delta t_{x,j}; w_{x,j}) (H_{ex}(\Delta t_{x,j}; w_{x,j}) + \\
&\sum_{i \in c, t_i < t_j} H_{in}(\Delta t_{i,j}; w_{i,j}))
\end{aligned}
\tag{1}
$$

The infection of nodes are independent with each other, so the joint distribution of infection events happening in one cascade c is $f_c(t_c; W) = \prod_{j \in c} f_j(t_j; W)$. In most cases, we observe several cascades of different information and we define the set of cascades as set Q, then the likelihood of all cascades is the product of the likelihoods of each individual cascade $f_Q(t_Q; W) = \prod_{c \in Q} f_c(t_c; W)$. Note that because this is a generative model, we must suppose a distribution of prior probability $p(W)$ of different network structure where W is just the matrix consisting of each edge weight $w_{i,j}$(including $w_{x,j}$) and $f_{in}(\Delta t_{i,j}; w_{i,j})$ and $f_{ex}(\Delta t_{x,j}; w_{x,j})$. We think each network structure is of same possibility, that is, $p(W)$ is a uniform distribution, and thus according to Bayes' Theorem, maximum likelihood(ML) problem $f(W|Q) = \frac{f_Q(t_Q; W) p(W)}{\sum_W f_Q(t_Q; W) p(W)}$ changes into maximum a posterior(MAP) problem $f(W|Q) \to f_Q(t_Q; W)$. As for the distribution of $f_{in}(\Delta t_{i,j}; w_{i,j})$ and $f_{ex}(\Delta t_{x,j}; w_{x,j})$, we can apply different or same distributions to these two. Exponential, power-law and Rayleigh distribution(Fig.2) are the common

choices for these two distributions where $\alpha_{i,j}$ in the table is just $w_{i,j}$. Next we use MLE to estimate the best possible parameter matrix W.

Model	Transmission likelihood $f(t_i\|t_j;\alpha_{j,i})$		Log survival $\log S(t_i\|t_j;\alpha_{j,i})$	Hazard $H(t_i\|t_j;\alpha_{j,i})$
Exp	$\begin{cases}\alpha_{j,i}\cdot e^{-\alpha_{j,i}(t_i-t_j)}\\0\end{cases}$	if $t_j<t_i$ otherwise	$-\alpha_{j,i}(t_i-t_j)$	$\alpha_{j,i}$
Pow	$\begin{cases}\frac{\alpha_{j,i}}{\delta}\left(\frac{t_i-t_j}{\delta}\right)^{-1-\alpha_{j,i}}\\0\end{cases}$	if $t_j+\delta<t_i$ otherwise	$-\alpha_{j,i}\log\left(\frac{t_i-t_j}{\delta}\right)$	$\alpha_{j,i}\cdot\frac{1}{t_i-t_j}$
Ray	$\begin{cases}\alpha_{j,i}(t_i-t_j)e^{-\frac{1}{2}\alpha_{j,i}(t_i-t_j)^2}\\0\end{cases}$	if $t_j<t_i$ otherwise	$-\alpha_{j,i}\frac{(t_i-t_j)^2}{2}$	$\alpha_{j,i}\cdot(t_i-t_j)$

Fig. 2.

C. Proposed Solution

The final form of the problem is as follows:

$$\begin{aligned} &min_W\, L(W) = -\sum_{c\in Q} log(f_c(t_c;W)) \\ &subject\,to\, w_{i,j}\geq 0, for\, all\, i,j\in V'=V\cup x \end{aligned} \tag{2}$$

It is easy to verify that the constraints are convex sets. The object function is also convex because it only consists of *log* function, hazard function *H* and survival function *S*. We know that *log* function, hazard function and survival function are all concave, and the combination and addition of these three functions are still concave, so the negative of a concave function turns into a convex object function. We can use many shelf-off tools to solve this form of MLE problem using the method like stochastic gradient descent or Newton method. To save time, if node *i* or *j* do not appear in any cascade, we set $w_{i,j}=0, w_{j,i}=0, w_{x,j}=0, w_{x,i}=0$ since node *i* or *j* are not infected by the information and thus we do not have enough evident to infer the edges between them. For every node *j*, we set $w_{j,x}=0$ because it is intuitive that the information spreads from mass media to the social networks while the reverse direction makes no much sense. Otherwise iterate the following formula until convergence or $w_{j,i}=0$:

$$w_{j,i}^{(k+1)} = w_{j,i}^{(k)} - \alpha\Delta_{w_{j,i}} L(W^{(k)}) \tag{3}$$

This is just the method of gradient descent. Because we have take Naive Bayes assumption that different parameters $w_{i,j}$ are independent from each other, so the optimization problem can split into many several subproblems and we can parallelly compute different parameters which reduces the time complexity much.

III. Further Modifications

Previously we assume $w_{j,i}$ is same for different cascades while this is often not the case. Our friends actually can be separated into different groups. Some of them have common interests with us. For example, they also like sports or movies. Some of them may be our research partners. Then the information about research may spread quickly among the latter group while the information about sports may diffuse quickly among the former group. In a word, we have different kinds of relationships with different people and this kind of differences can be reflected by the difference of spreading pattern between different cascades. Also, our relationships vary along with time. Some may decay till disappear. Just consider the case of the relationship between us and our primary school classmates. Some may become stronger or new connections are built. Some may also change the form. For example, you may talk about topics about sports with your research partner in occasion and you surprisingly find that he is also a sports fan just like you. Then your relationship is not an academic one but also becomes a private one and he belongs to both groups of friends. To take all these situations into consideration, we change the parameter $w_{j,i}$ into $w_{j,i,c,t}$ which means that $w_{j,i}$ relies on the content of cascade *c* and the time *t* of the information diffusion process.

This is a really complex model, but we can use the idea of discretization to simplify it. To clear the subscript variable *c* from $w_{j,i,c,t}$, we can divide many different information into just a few groups such as entertainment, academy and etc. according to the content. Then for each group, the problem is the same as before with variable *c* disappearing and we can learn different parameter matrix W for different groups simultaneously. As for the subscript variable *t*, we can give more weights to the parameter $w_{j,i,t}$ inferred by the latest cascade and then take weighted average on all $w_{j,i,t}$ to clear the variable *t* and get the final $w_{j,i}$. This model can be rewritten as $w_{j,i,t} = m_{j,i}w_{j,i}$ where $m_{j,i}$ just represents the relative weight of parameter $w_{j,i}$. To calculate $m_{j,i}$, the simplest way is to suppose a same time decay model for all edges but this wrongly assumes that all edges decay and decay to the same extent when time passes by. A better way is to combine $m_{j,i}$ to form a transformation matrix M to represent the dynamics changes in the network. We suppose the network only changes a little in a certain period of time while changes a lot between these time periods. Without losing generality, just assume this period of time is a month. Then we infer parameter matrix W_t for a month, using cascades happening in that month and infer parameter matrix W_{t+1} for the next month. These two parameter matrix satisfy $W_{t+1} = MW_t$. Hence, the transformation matrix *M* can be calculated by $M = W_t^{-1}W_{t+1}$. Repeat this procedure and take average to revise the original matrix *M* to get the latest one. If we want to predict the network structure in the next year, we can just calculate it by $W_{next-year} = M^{12}W_{this-year}$ since a year has 12 months.

IV. Conclusion of my contribution

Based on the previous probabilistic model on information propagation, I do several modifications:

a) The original model only considers information diffusion inside the network but ignores external influences while I

consider the external influences by introducing an external source node *x*, new edges *edge(x)* and new function $f_{ex}, F_{ex}, S_{ex}, H_{ex}$ to the original model.

b) The original work does not consider the effects of difference of content between different cascades on the parameters while I consider that by replacing $w_{j,i}$ with $w_{j,i,c,t}$ and then clear variable *c* by separating information into just a few groups and running the original algorithm simultaneously for these few groups.

c) The original model assumes all relationships decay and decay to the same extent when time passes by while I consider the more general case using a transformation matrix *M* and to learn this matrix by using data(cascades) from two successive period of time. Then use this matrix *M* to predict the network structure in the future.

d) The original model also sets a window size *T* which increases the model complexity but I think it does not make much sense so I remove it off for simplicity because if some nodes keep uninfected when the information diffusion is going to stop, they are really unlikely to be infected after the observing time *T*, that is, an ended information diffusion process is not likely to revive and become popular again.

V. Future Work

Due to the limit of resource and time, I have not experimented my model in real data yet. Also, I suppose the network almost remains static in a period of time and only changes between time periods, which is a relative simple hypothesis about the dynamic changes in the network. Recently, a paper about the Bursty Dynamics of the Twitter Information Network published by Seth Myers and Jure Leskovec talks about the abrupt changes in the network and finds that the abrupt changes are often accompanied by information diffusion process. Combining with their ideas, I may propose a more robust model under the dynamic situation. In short, my following work include these two parts:

a) Continue programming and debugging the model and then test it on the synthetic data and real data.

b) Deeper research on dynamic networks, such as abrupt changes(burst) in networks using Seth Myers and Jure Leskovec's idea.

Acknowledgment

I would like to thank Prof.Wang and Prof.Tian for their teaching and help inside and also outside the field of academy, especially Prof.Wang for the great help to introduce me into his big data group where I have learned a lot.

I would also like to thank TA Chuan Ma for his patience and tolerance in the lab, homework and report part. I still remembered when our group meeted a great problem doing lab4 and became upset, Mr.Ma instructed us how to find the solution with great patience.

Finally, I would like to thank all members in Big Data Group for their kindness and warm-heart, especially Songjun Ma, Ge Chen, Shiyu Liang and Ruotian Luo for their great help when I first came into this group.

[1] [2] [3] [4] [5] [6]

References

[1] M. G. RODRIGUEZ, J. LESKOVEC, D. BALDUZZI, and B. SCHÖLKOPF, "Uncovering the structure and temporal dynamics of information propagation," *Network Science*, vol. 2, no. 01, pp. 26–65, 2014.

[2] M. G. Rodriguez, D. Balduzzi, and B. Schölkopf, "Uncovering the temporal dynamics of diffusion networks," *arXiv preprint arXiv:1105.0697*, 2011.

[3] M. Gomez Rodriguez, J. Leskovec, and A. Krause, "Inferring networks of diffusion and influence," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 1019–1028.

[4] D. Wang, H. Park, G. Xie, S. Moon, M.-A. Kaafar, and K. Salamatian, "A genealogy of information spreading on microblogs: A galton-watson-based explicative model," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 2391–2399.

[5] S. A. Myers and J. Leskovec, "The bursty dynamics of the twitter information network," in *Proceedings of the 23rd international conference on World wide web*. International World Wide Web Conferences Steering Committee, 2014, pp. 913–924.

[6] S. A. Myers, C. Zhu, and J. Leskovec, "Information diffusion and external influence in networks," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2012, pp. 33–41.