Using Survival Theory in Early Pattern Detection for Viral Cascades

Xiaofeng Gao*, *Member, IEEE,* Xiaosong Jia, Chaoqi Yang, and Guihai Chen, *Senior Member, IEEE*
E-mail: {gao-xf, gchen}@cs.sjtu.edu.cn, jiaxiaosong@sjtu.edu.cn, ycqsjtu@gmail.com.

Abstract—In recent years, social networks have developed rapidly and become an indispensable part of people's everyday life. There are many models trying to predict whether some reshare cascades are going to be popular or not. But most of the models' performances are limited due to the lack of cascades' information in the early pattern.

In this paper, we propose Early Pattern detection model for Outbreak Cascades(in abbreviation, EPOC) inspired by the survival theory. We use three features to predict cascades' virality: retweet sequence, follower number sequence and timestamps of the first tweet which includes both the static and dynamic characteristics of cascades. We utilize the theory that distributions of both viral and non-viral cascades are Gaussian to get the boundary between these two kinds of cascades with sufficient proof to testify its rationality. To detect the virality more precisely and earlier, we utilize hazard functions in the survival theory which can capture the bursting of the cascades and propose two different hazard ceilings. We do a series of experiments to analyze impacts of different factors to performance of our model measured by three practical metrics. The results shows that our model is relatively static and outperforms several state-of-art baselines.

Index Terms—Early-Stage Detection, Outbreak Cascade, Survival Theory, Cox's Model, Social Networks

1 Introduction

In recent years, people's everyday lives have changed a lot due to tremendous development of technology. One typical instance is that social networks like Twitter and Weibo has become an indispensable part of people's daily life. Users of these online social platforms can *tweet* short messages (e.g., up to 140 characters in Twitter) to express their feeling and others can give likes, leave comments or *retweets*. And the retweet could potentially disseminate and further spread information to a large number of users, which forms a *cascade* [1]. But most of the cascades would spread on a small scale and finally be stable at a relatively small size. And only a small portion of cascades would become popular and have a tremendous influence which we call *viral*. When these cascade grows larger and there are more people involved, a sudden *burst* will definitely arrive, which we call a *spike*. In fact, detecting and predicting the burst pattern of a cascade, especially at early stage, attract lots of attention in various domains: meme tracking [2], stock bubble diagnosis [3], and sales prediction [4], etc.

However, to fully understand the burst pattern of cascades ahead of time will meet three major challenges:

- **First**, it is hard to catch some effective signs indicating whether or not one cascade will burst because of cascades' disorderly increasing and bursting manner at early stage and lack of information [5].
- **Second**, cascades have so different life spans [6] that extracting distinguishable features is difficult. Additionally, various life spans usually mean researchers often have trouble when setting suitable observation time.
- **Third**, the burst pattern of cascades usually follows a quick *rise and fall* law [7], which lasts shortly but can make huge influence. Traditional approaches of cascades prediction and detection are mostly suitable to common ideal cases without considering sudden rise and fall, in which situation it is reasonable to rely on historical information to build the near future. Nevertheless, with this sudden rise and fall situation, the correlations between the history and the near future can be hardly caught by existing traditional models.

As **firstly** shown in Fig.1a, we plot the diffusion process of seven real-world cascades from Twitter. We can see that @Cascade2 shares almost the same pattern with @Cascade1 before it outbreaks at time t_0, which means that it is hard for us to catch the distinguishing signs using the early information. As the **second** challenge states, @Cascade1∼7 represent different life span at early stage. While @Cascade6 ends its diffusion, @Cascade3 is just about to start propagation, and it still enlarges even at the end of observation. The **third** challenge can be vividly described in Fig.1b, where we focus on @Cascade2 and plot how it is retweeted. Fig.1b shows that @Cascade2 experiences a mild propagation when it appears, but after time t_0, it goes through two large retweeting spikes (sudden falls in survival curve plotted in Fig.1c), and the final amount of retweeting explodes to about 1600 during the burst period.

These three core challenges motivate us to design a model that can handle this quick rise and fall pattern, characterize different cascades uniformly, and detect the burst pattern as early as possible.

Motivated by the study of death in biological organisms, in this paper, we regard the diffusion of cascades as the

- *X. Gao* (Corresponding Author), X. Jia, C. Yang and G. Chen are with the Shanghai Key Laboratory of Scalable Computing and Systems, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, P.R.China 200240.*

Manuscript received April 19, 2005; revised August 26, 2015.

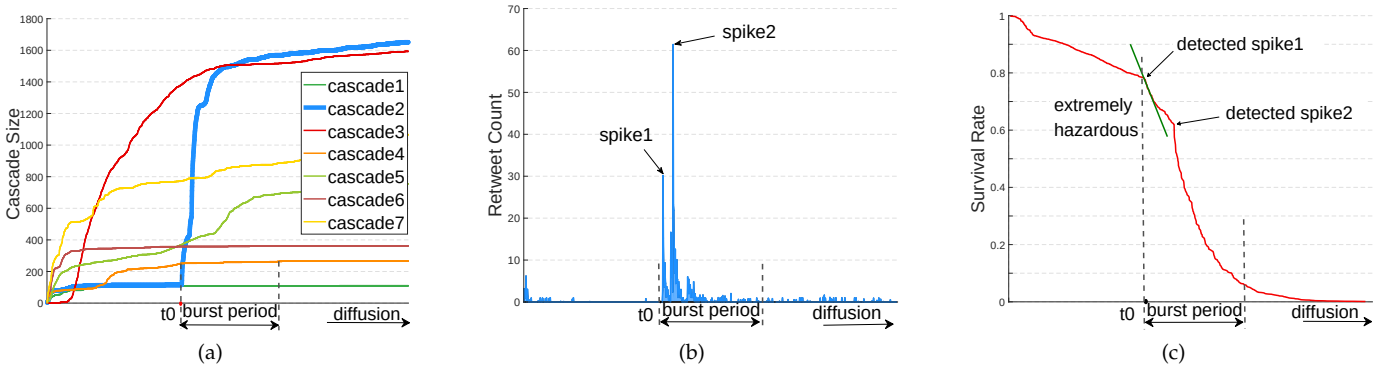


Fig. 1. Samples of Cascade Diffusion on Twitter. (a) Cascade Life Cycle; (b) Retweeting @Cascade2; (c) Survival Curve @Cascade2.

growing process of biological organisms. Since Cox's model is widely used to characterize the life span of biological organisms, here we adopt Cox's model with the knowledge of cascades, transforming the burst detection task into diagnosis of cascade life table, and then we build a survival perspective Early Pattern detection model for Outbreak Cascades, in abbreviation, *EPOC*. Though previous work [8] has also tried Cox's model, their work is mainly based on unsubstantiated observations as well as only taking one feature into consideration, which does not address all the above challenges at all.

In our *EPOC*, to consider the influential factors from different perspectives, we harness three features from each cascade (*retweet sequence*, *follower number sequence*, and *original timestamps*) to capture the effectiveness of temporal information [9], the influence of involved users [10], [11], and the dynamics of user activity [12]. **Then**, to study the distinctiveness of cascades' life span, we train an effective Cox's model and employ two Gaussian distributions to fit the survival probability of viral and non-viral cascades at different time point respectively, and obtaining a survival boundary between the viral and the non-viral, which is further proven to be well-defined theoretically. **Finally**, as the static and dynamic nature of cascade diffusion are both important indicators of cascade virality, we jointly consider survival probability and hazard rate, which considerably enhances our model's performance in handling the quick rise and fall pattern. We then employ three special metrics (K-coverage, Cost, Time ahead) to compare *EPOC* with two basic machine learning methods (LR, SVR) and three powerful s published in recent literatures (PreWhether [13], Seismic [10],SansNet [8]) on two large real-world datasets: Twitter and Weibo. Experiment results show that *EPOC* outperforms these five methods in burst pattern detection at very early stage.

Our main contributions are summarized as:

- We apply the survival theory into our model with rigorous theoretical analysis and establish a powerful burst detection model EPOC for cascade diffusion, which can handle the quick rise-and-fall pattern as well as the significantly distinct life span of cascades at the early stage.
- We utilize both static and dynamic information from cascades and obtain a dimidiate boundary with two Gaussian distribution for viral and non-viral cascades.
- We novelly use the burst pattern to help predict the popularity of an online content with two different boundaries which have different advantages.
- We adopt three special metrics and conduct extensive experiments on two large real-world data sets (Twitter and Weibo) with different parameters. We make in-depth analysis of the effects of those parameters and the results show that EPOC gives the best performance comparing with several state-of-the-art approaches.

The remainder of the paper is organized as follows. Section 2 introduces some basic information about this field and several related works. Section 3 gives some common notations about cascade prediction. Section 4 makes a brief review of the survival theory. And we also explain the advantages of adopting it and how to apply it into our viral cascades detection model. Section 5 demonstrates our model EPOC in details and gives rigorous theoretical analysis about it. In Section 6, we do experiments on two datasets: Twitter and Weibo and compare our model to several baselines. Further, we also do experiments on two different parameters and analyze their influences on the results. Finally, we conclude our work and highlight the possible future perspectives in Section 7.

2 Related Work

In recent years, social networks have successfully attracted researchers attention, and plenty of achievements have been made in the past few decades, especially when it comes to the study of information cascades, including the prediction of cascade size, how a cascade grows and disseminates.

2.1 Information Cascade and Social Networks

The study of information cascades has been going for a long time, and it is of great use in many applications, such as meme tracking [2], stock bubble diagnosis [3], sales prediction [4], e-mails [14], product recommendation [15] and website [16]. In recent years, works probing into cascades emerge. The literature concerning cascade in social networks can be divided into three categories [17], [18]. The first category lays on user level prediction. One of the pioneers is Iwata et al. [19], they propose a Bayesian inference model with stochastic EM algorithm, trying to discover the latent influence among online users. [20] also utilizes user-related features to help social event detection. [21] concludes that the largest cascades tend to be generated

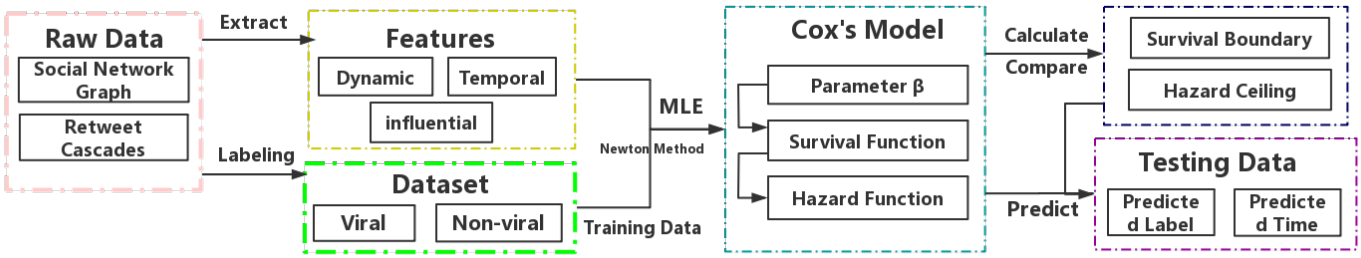


Fig. 2. Flow chart of *EPOC* model. First, we extract three kinds of features (dynamic, temporal, influential) from raw data as well as labeling all the cascades in the dataset. Then, we use these features and labels in training dataset to train a Cox Extended Survival Model by maximum likelihood estimation and Newton method. Next, we use this survival model to obtain survival boundary and hazard ceiling. Finally, we use the survival model, survival boundary and hazard ceiling to predict the cascades in the testing set.

by influential users with a large number of followers. [22], [23], [24], [25], [26], [27] focus on the behavior of individual users to predict the resharing of tweets or URLs.

Additionally, some other researchers also analyze the topology, since structural feature is said to be one of the predictors of cascade size [28]. PageRank of retweeting graph is taken into consideration [29], [30], while [31] utilizes the number of directed followers as one of the important infectors. Apart form static network structure, [32] also explores the strength of every edge in the diffusion of cascades.

The association between cascades and temporal information has also been widely mined. Many experimental results, such as [9], [10], reveal that temporal features are the most effective type of indicators. [7] is one of the first to utilize temporal features, and they implement SpikeM, which is able to generate all the possible rise-and-fall patterns of cascades. To depict the connection between early cascade and its final state, both [5] and [13] propose Bayesian networks with temporal information. Other temporal information, like mean time and the maximum length of all time intervals, has also been considered [9].

2.2 Outbreak Detection and Modeling

Burst or outbreak, defined as "a brief period of intensive activity followed by long period of nothingness" [6], is a common phenomenon during the diffusion of social content, which is worthy of studying and may bring benefits to modern society. Existing works probing into cascades mainly focus on prediction of its future popularity [5], [13], [29] or final aggregate size [10], [28]. However, how to detect the burst pattern of large cascade in early stage remains an intriguing problem. Recently, based on the transformation of time window, Wang et al. [6] proposes a classification model to predict the burst time of cascade. Unfortunately, their approach acquires laborious feature extraction, and the traditional classifiers they used can hardly take the best use of the features. [33] identifies social bursts by considering the spreading effect of social bursts in the spatial-temporal contexts. [34] propose a user burst topic graph model which can represent topology structure of topics' propagation. [35] and [36] implement logistic models, which consider all the nodes as cascade sensors. Just as bad, when the number of nodes in networks turns to be billions, these implementations will be particularly difficult.

2.3 Survival Theory and Its application in Prediction

Survival theory [37], [38] is proposed to make use of the lifetime data to infer the unknown regression coefficients in the medical statistics field, such as death in the biological organisms, diagnosis of the cancer or failure in a medical methods. Since it can model the time series data well, it has been widely applied into many other areas like engineering(reliability analysis) [39], economics(duration analysis) and sociology(history analysis).

Recently, researchers have adopted it to predict the information cascades [8], [40], [41]. [40] uses survival analysis to capture the three features of information diffusion. To predict the ultimate cascade, [41] develops a network inference model with additive or multiplicative risk. One drawback is that their training set contains 80% records of cascade, we deem the observation is too long, besides either observing the whole network or inferring them is a tough problem [8]. SansNet is proposed in [8], predicting whether and when a cascade goes viral. This approach utilizes only the size of cascades as feature, making it weak to apply to multiply cases, since the features of an author [22] and the inherent network [28] are sometimes more important than features from cascade itself [22]. Another drawback of this approach is that survival curve cannot totally reveal status of cascades.

In this work, adopting survival theory, we can exactly overcome all of the drawbacks mentioned from the perspective of cascade dynamics. Utilizing early features from both cascades and users, as well as employing hazard rate into inference, experiment shows that our *EPOC* can effectively detect the early pattern of outbreak cascades.

3 Problem Statement

In this section, we will give some formal definitions of the viral cascades detection problem.Table 1 gives notations used in this paper.

Initially, when a user shares the content with her set of friends, several of these friends share it with their respective sets of friends, and a *cascade* of resharing can develop [28]. So we have Definition 1 for cascade.

Definition 1 (Cascade). *We dene a **cascade** as a set of ordered time when retweetings happen as* $C = \{t_1, t_2, ..., t_n\}$ *with its **feature** matrix* $X = \{\mathbf{x_1}, ..., \mathbf{x_n}\}$ *where* t_0 *denotes the time when the origin tweet posted,* $t_i (2 \leq i \leq n)$ *means* i^{th} *retweeting happened at time* t_i *and* $\mathbf{x_i} (\mathbf{1} \leq \mathbf{i} \leq \mathbf{n})$ *denotes the feature vector of the* i^{th} *retweeting.*

TABLE 1
Notations in Our Paper

Name	Description
D	Dataset which contains k cascades
F	Feature matrices set of dataset
r	Virality ratio, a hyper parameter between 0 and 1
C_k	k^{th} cascade which contains a series of time
X_k	Feature matrix of k^{th} cascade
n_k	Length of k^{th} cascade
$t_{k,i}$	i^{th} time in the k^{th} cascade
$\mathbf{x_{k,i}}$	Feature vector of i^{th} time in the k^{th} cascade
ρ	Length threshold for viral cascades
D_s	Time to event data obtained from D
T	Random variable for survival time of a individual
$f_i(\cdot)$	Density function of variable T of individual i
$F_i(\cdot)$	Cumulative distribution function of individual i
$S_i(\cdot)$	Survival function of individual i
$h_i(\cdot)$	Hazard function of individual i
$H_i(\cdot)$	Cumulative Hazard function of individual i
$S^*(\cdot)$	Survival boundary obtained from training data D
$h_\alpha(\cdot)$	Hazard ceiling of EPOC model

Normally, when measuring the popularity of a tweet, we refer to the total number of retweeting about this tweet, i.e. length n of C. Then we define viral cascade in Definition 2.

Definition 2 (Viral Cascade)**.** *For a given threshold ρ and a cascade $C = \{t_1, t_1, t_2, ..., t_n\}$, if $n > \rho$, we call that cascade **viral**, and otherwise **non-viral**.*

We usually use a relative threshold like top 5 % longest cascade's length, considering the fact that only a few cascades can be hot on the Internet.

But due to the very limited length of observing time window, we often cannot get when exactly the size of a cascade stop growing and the final size of it, which means maybe we cannot know whether or not a cascade is viral directly. So, we turn to predict whether and when a cascade will become viral in the future. This prediction task is defined in Definition 3.

Definition 3 (Viral Cascade Prediction)**.** *Given a set of k cascades time series data $D = \{C_1, C_2, ..., C_k\}$ within the watching window with these cascades feature matrices $F = \{X_1, X_2, ..., X_k\}$, we want to predict each cascade is either viral or non-viral. Furthermore, for a predicted viral cascade, we want to predict an exact time t when this cascades size begins to be larger than the threshold ρ.*

4 Survival Theory and Cox's Model

In this section, we introduce some basic knowledge about survival theory and Cox's model and describe how we are inspired by them and apply them into *EPOC* model.

4.1 Survival Theory

Survival theory is a kind of statistical techniques which are used to model time to event data. To begin with, we use the term **event** to indicate that what we are interested in. The time to event data can be like: $D_s = \{t_1, t_2, ..., t_k\}$ with $F = \{X_1, X_2, ..., X_k\}$ where t_i denotes the event happens to individual i at time t_i and X_i denotes the features of individual i which can be either time-dependent or time-independent. We also use the term **failure** to indicate the occurrence of the target event and the term **survival time** to define the time for failure to happen.

In our *EPOC* model, *event* is that a cascade's size becomes larger than threshold, i.e. that cascade becomes viral, **failure** of a cascade means this cascade becomes viral and **survival time** t_i means i^{th} cascade becomes viral at time t_i.

Note that in our problem, each t_i in D_s is corresponding to one cascade while each $t_{k,i}$ in C_k is corresponding to one retweet in this cascade. In other words, all the cascades' failure time makes D_s.

Let T be a positive random continuous variable for the survival time of a individual. Then we denote f(t) as its density function and $F(t) = P(T < t)$ as its cumulative distribution function in Definition 4.

Definition 4 (Survival Function)**.** *Normally, we define the survival function as:*

$$S(t) = P(T \leq t) = 1 - F(t) = \int_t^{+\infty} f(x)\mathrm{d}x \tag{1}$$

As we can see in (1), S(t) gives the probability that the individual survive to the time t.

Then we want to get the instantaneous probability of failure which is defined in (2).

Definition 5 (Hazard Function)**.** *We define the hazard function as:*

$$h(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta | T \geq t)}{\Delta t} \tag{2}$$

This formula indicates: giving the failure does not happen at time t, we want to get the conditional probability that the failure happens in $[t, t + \Delta t)$.

Since $\frac{P(t \leq T < t+\Delta | T \geq t)}{\Delta t} = \frac{P(t \leq T < t+\Delta, T \geq t)}{P(T \geq t)} = \frac{P(t \leq T < t+\Delta)}{P(T \geq t)}$, we can directly get (3) from (1) and (2).

$$h(t) = \frac{f(t)}{S(t)} = -\frac{\mathrm{dS(t)}}{\mathrm{dt}} \frac{1}{S(t)} = -\frac{\mathrm{d}}{\mathrm{dt}} \ln S(t) \tag{3}$$

In other words, we have (4).

$$S(t) = \exp(-\int_0^t h(x)\mathrm{d}x) \tag{4}$$

Then we want to get the sum of the risks the individual face since the observation begins:

Definition 6 (Cumulative Hazard Function)**.** *we define cumulative hazard function in (5).*

$$H(t) = \int_0^t h(x)\mathrm{d}x \tag{5}$$

4.2 Censored Mechanism

One advantage of survival theory is that it can deal with censored data, which means it can make use of data from the individuals whom events do not happen to during the observing time window. Assume that we have k individuals with survival function S(t), density f(t) and hazard rate h(t). Assume that individual i is observed until t_i, i.e., either events happen to it at t_i so that we do not need to observe anymore, or t_i is the end of the observation window and we call this situation censored.

For the former one, it contributes to the likelihood function with the term (6).

$$L_i^{death} = f(t_i) = S(t_i)h(t_i) \tag{6}$$

i.e. it survived until t_i and then died.

For a censored one, its contribution term is (7).

$$L_i^{censored} = S(t_i) \tag{7}$$

i.e. all we know is that it survives at time t_i and we do not know what will happen in the future.

In the survival analysis, all the data can make contribution to the final result without having to make any strong assumptions or delete any data, which means it have more information than some traditional approaches and achieve higher accuracy. And this advantage can be especially suitable for our viral cascades prediction task since a huge part of cascades do not become viral during observation window, in other words, censored.

4.3 Cox's model and Likelihood Function

Now after defining the survival problem in details, we will introduce Extended Cox Proportional Hazards Model for Time-Dependent Variables [38] to capture the effects from the input variables on survival time.

In Cox's model, the hazard function at time t for an individual i with features $\mathbf{x_i}(t)$ is defined in (8).

$$h_i(t) = h_0(t)exp(\boldsymbol{\beta^T X_i}(t)) \tag{8}$$

where $h_0(t)$ is the baseline hazard function which indicates the prior failure possibility at time t. Later we will show that it does not matter for prediction task and is mainly set to keep the hazard function in the interval [0,1] and monotonically decreasing. The term $exp(\boldsymbol{\beta}^T\mathbf{X_i}(t))$ is the relative risk which indicates the effect of feature variables to survival possibility and can be not monotonically decreasing.

Since the model is proportional and we just want to compare hazard possibilities between cascades, we obtain the relative hazard rate $\lambda_{i,j}$ in the following concrete way as shown in (9).

$$\begin{aligned}\lambda_{i,j} = \frac{h_i(t)}{h_j(t)} &= \frac{h_0(t)\cdot \exp\left(\boldsymbol{\beta}^T \boldsymbol{X_i}(t)\right)}{h_0(t)\cdot \exp\left(\boldsymbol{\beta}^T \boldsymbol{X_j}(t)\right)} \\ &= \frac{\exp\left(\boldsymbol{\beta}^T \boldsymbol{X_i}(t)\right)}{\exp\left(\boldsymbol{\beta}^T \boldsymbol{X_j}(t)\right)}\end{aligned} \tag{9}$$

where β is the parameter vector, $\mathbf{X_i}(\mathbf{t})$ and $\mathbf{X_j}(\mathbf{t})$ are respectively the feature vectors of i^{th} and j^{th} cascade. From the equation, it is easy to conclude that the baseline hazard does not play any role in relative hazard rate $\lambda_{i,j}$. Therefore, instead of considering absolute hazard function, we only care about the relative hazard rate of cascades, which only concerns parameter vector $\boldsymbol{\beta}$.

Next, we use *Maximum Likelihood Estimation* to get parameter vector $\boldsymbol{beta}$. We denote i^{th} cascade's time-to-event as t_i, and assume that $0 < t_1 < t_2 < \cdots < t_n$. Then the Cox's partial likelihood is given by (10).

$$\begin{aligned}L(\boldsymbol{\beta}) &= \prod_{i=1}^{n}\left(\frac{h_i(t_i)}{\sum_{j=i}^{n} h_j(t_i)}\right)^{\delta_i} \\ &= \prod_{i=1}^{n}\left(\frac{exp\left(\boldsymbol{\beta}^T X_i(t_i)\right)}{\sum_{j=i}^{n} exp\left(\boldsymbol{\beta}^T X_j(t_i)\right)}\right)^{\delta_i}\end{aligned} \tag{10}$$

where δ_i depends on the cascade termination state, i.e., if the event happens to i^{th} cascade at t_i, then δ_i equals to 1, and otherwise it is censored and δ_i equals to 0.

The explanation of the model is that: at time t_i, the individual should have the largest possibility to fail or censor. By maximizing the fraction with the hazard rate of individual i as numerator and the sum of hazard rates of all alive individuals as denominator, it approximately matches the fact that individual i failed or is censored at this point. In addition, we can see that in the likelihood function $h_0(t)$ is eliminated again which shows that it will not influence the result.

Thus, the log-partial likelihood of parameter vector $\boldsymbol{\beta}$ can be calculated as shown in (11)

$$\log L(\boldsymbol{\beta}) = \sum_{i=1}^{n}\delta_i\left[\boldsymbol{\beta}^T X_i(t_i) - \log\left(\sum_{j=i}^{n}\exp\left(\boldsymbol{\beta}^T X_j(t_i)\right)\right)\right] \tag{11}$$

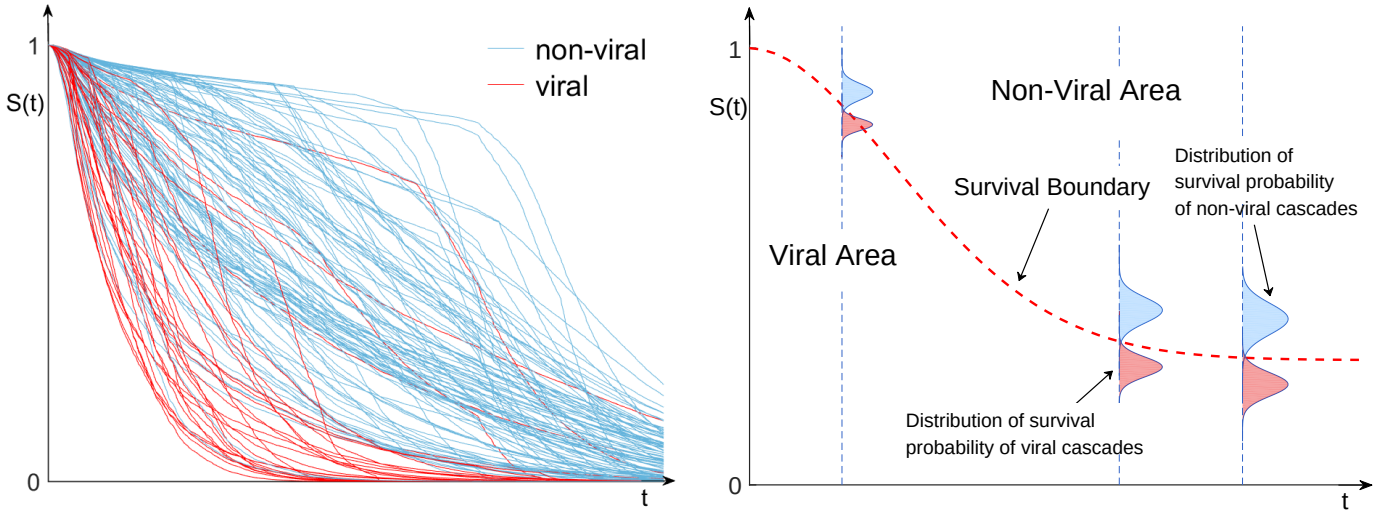
In order to maximize the log-partial likelihood by using Newton method with $\frac{\mathrm{d}\log L(\boldsymbol{\beta})}{\mathrm{d}\boldsymbol{\beta}}$, we can get the numerical estimation of parameter vector $\boldsymbol{\beta}$.

5 EPOC: Detecting Early Pattern of Outbreak Cascades

Based on the basic model stated previously, in this section, we combine the Cox's model with our knowledge of cascades, and make it suitable to handle the task of detecting the early pattern of outbreak cascades. Here we regard cascades as complex dynamic objects that pass through successive stages as they grow. During this process of growth, the survival probability and the hazard rate of cascades will change dynamically. High survival probability and low hazard rate suggest that cascades are unlikely to be viral in the future, while low survival probability as well as high hazard rate implies the opposite. In this sense, we introduce *survival boundary* and *hazard ceiling* to help accomplish this challenging task at very early stage.

5.1 Feature Selection

We firstly consider the **timestamp of each retweet** of cascades, because temporal information is regarded as the most effective indicator revealing the diffusion of cascades [9], [10]. Besides, [21] concludes that the users can have a great influence in the growth of cascades, thus **the number of followers** of every spreader involved in a cascade is also taken into consideration. Additionally, users are more active during the daytime than in the midnight [12], owing to this variation and dynamics of user activity, our model also includes **the timestamp of the first tweet**.



(a) (b)

Fig. 3. (a) Survival Functions of Cascades: the red lines represent the survival functions of viral cascades, and the blue lines show the non-virals'; (b) Survival Boundary: the red dashed line separates the two categories of blue (non-viral cascades) and red (viral cascades).

5.2 Survival Boundary: a Static Perspective

The first step of our model is to use all the training data to get the parameter $\boldsymbol{\beta}$. Then, in order to detect the early pattern of outbreak cascades, we characterize the survival functions of all cascades. Shown in Fig.3a, the red lines represent the survival functions of viral cascades, and the blue lines show the non-virals'. Then we are supposed to divide estimated survival functions of all cascades into two classes (viral and non-viral). In other words, we need to find a survival boundary. As is illustrated in Fig.3b, the red dashed line separates the two categories of blue (non-viral cascades) and red (viral cascades).

Previous works [42] have demonstrated that at a fixed observing time t, the distribution of survival probability of different cascades obeys Gaussian distribution. Based on this knowledge, we employ two random variables: f_v^t (for viral cascades) and f_n^t (for non-viral cascades) subject to time t, which satisfy the Gaussian distribution.

Formally, we specify this assumption in Definition 7

Definition 7 (Gaussian Distribution Assumption). *For any Given time t, we have $f_v^t \sim \mathcal{N}(\mu_v^t, \sigma_v^t)$ and $f_n^t \sim \mathcal{N}(\mu_n^t, \sigma_n^t)$, where μ_v^t, σ_v^t and μ_n^t, σ_n^t are the parameters of Gaussian distribution for viral and non-viral cascades subject to time t.*

Based on Definition 7, for a given time t, the survival probability of viral and non-viral cascades can be respectively characterized as f_v^t and f_n^t . Therefore, the task to find the optimal survival boundary function is to give the suitable separation between the two Gaussian distributions at any given time t.

Lemma 1 shows how we use the Gaussian distribution assumption to get the survival boundary function in (12):

Lemma 1 (Survival Boundary). *For any given time t, assume the survival boundary to be $S^*(t)$ which satisfies the following formula:*

$$\begin{aligned}&\int_{-\infty}^{S^*(t)} \frac{1}{\sqrt{2\pi}\sigma_v^t} \exp\left(-\frac{(x-\mu_v^t)^2}{2{\sigma_v^t}^2}\right) \mathrm{d}x \\ &= \int_{S^*(t)}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_n^t} \exp\left(-\frac{(x-\mu_n^t)^2}{2{\sigma_n^t}^2}\right) \mathrm{d}x\end{aligned} \tag{12}$$

Then the optimal survival boundary can be calculated from this equation as $S^*(t) = \frac{\mu_v^t \sigma_n^t + \mu_n^t \sigma_v^t}{\sigma_v^t + \sigma_n^t}$.

As is shown in Fig.4a, given time t, we plot the frequency histograms of survival probabilities of both viral and non-viral cascades (blue bars represent non-viral ones, and red bars represent viral ones). Then we use two Gaussian distribution curves f_v^t and f_n^t to fit these two histograms. Next, to simplify our problem, we employ the cumulative distribution function of f_v^t and f_n^t, respectively denoted as $F_v^t(s)$ and $F_n^t(s)$, specifically we have (13a) and (13b).

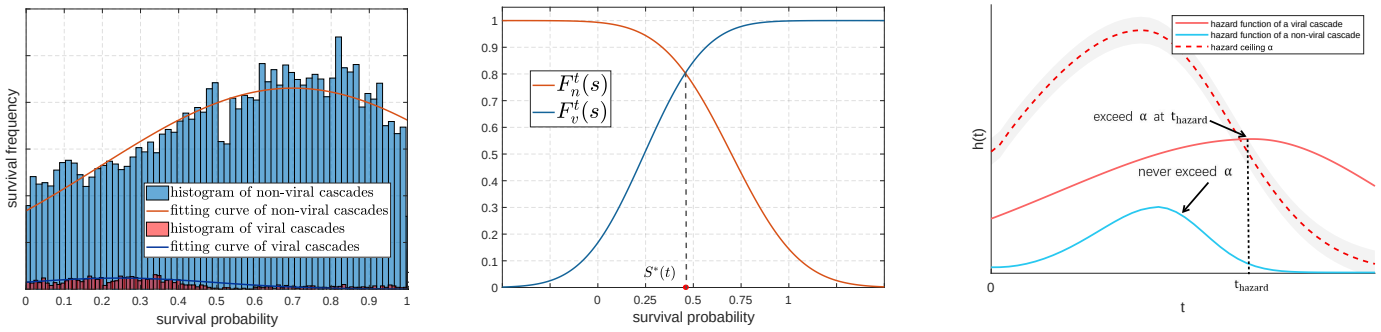
$$F_v^t(s) = P(S < s) = \int_{-\infty}^{s} \frac{1}{\sqrt{2\pi}\sigma_v^t} \exp\left(-\frac{(x-\mu_v^t)^2}{2{\sigma_v^t}^2}\right) \mathrm{d}x \tag{13a}$$

$$F_n^t(s) = P(S > s) = \int_{s}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma_n^t} \exp\left(-\frac{(x-\mu_n^t)^2}{2{\sigma_n^t}^2}\right) \mathrm{d}x \tag{13b}$$

Finally, we plot $F_v^t(s)$ and $F_n^t(s)$ in Fig.4b, and the x-coordinate of the only intersection $S^*(t)$ is the optimal survival boundary subject to time t.The intuition behind this boundary is that since it is x-coordinate of the only intersection point of the two Gaussian distributions, it will cause the same amount of *False Positive* mistakes and *False Negative* mistakes, which is natural and balanced.

5.3 Well-Definedness of Survival Boundary

In order to make the problem completer and more rigorous, in this subsection, we mainly discuss the monotonicity of the survival boundary, which is given in lemma 1, i.e., we



(a) (b) (c)

Fig. 4. (a) Survival Frequency at Time t: the frequency histograms of survival probabilities of both viral and non-viral cascades (blue bars represent non-viral ones, and red bars represent viral ones. Two Gaussian distribution curves fit two histograms correspondingly; (b) Survival Boundary $S^*(t)$ at Time t: curves of $F_v^t(s)$ and $F_n^t(s)$ and the x-coordinate of the only intersection $S^*(t)$ is the optimal survival boundary subject to time t; (c) Hazard Functions and Hazard Ceiling: the *hazard ceiling* is drawn in red dash line with a grey hazard-tolerant interval, and the red solid line and blue solid line respectively denote the hazard functions of a viral cascade and a non-viral cascade.

will prove that the optimal survival boundary in our model is also a survival function.

In fact, from the observing time window, we can conclude three solid facts:

- First, the survival probabilities of both viral and non-viral cascades are naturally monotonic decreasing with time t, so the average survival probabilities of both kinds of cascades are also monotonic decreasing.
- Second, non-viral cascades intuitively possess a higher survival probability. Thus, the average survival probability for non-viral cascades μ_n^t is reasonably larger than that of viral cascades μ_v^t.
- Third, real-word data shows that the range of non-viral cascades' survival probability appears to be more dynamic and uncertain than viral cascades, which means its relative fluctuation of standard deviation δ_n^t is also larger than viral cascades' standard deviation δ_v^t.

Formally, we specify these 3 conclusions in Lemma 2.

Lemma 2. *For any given time t, μ_v^t, σ_v^t and μ_n^t, σ_n^t respectively represent the average survival probability and its standard deviation of viral and non-viral cascades. Given time $t' > t$, we have the following conclusions:*

$$\begin{cases} \mu_v^t \geq \mu_v^{t'} \\ \mu_n^t \geq \mu_n^{t'} \end{cases}, \mu_n^t \geq \mu_v^t, \frac{\sigma_n^{t'} - \sigma_n^t}{\sigma_n^t} \geq \frac{\sigma_v^{t'} - \sigma_v^t}{\sigma_v^t}, \forall\ 0 < t < t' \tag{14}$$

Based on Lemma 1 and Lemma 2, we give detailed proof that the optimal survival boundary itself is a survival function in Lemma 3.

Lemma 3. *The optimal survival boundary $S^*(t)$ is monotonic decreasing with time t, i.e., $S^*(t)$ is also a survival function. Formally, we have*

$$S^*(t) \geq S^*(t'), \qquad \forall\ 0 < t < t', \tag{15}$$

Proof: For $\forall\ 0 < t < t'$, we have

$$\begin{aligned} & S^*(t) - S^*(t') \\ &= \frac{\mu_n^t \sigma_v^t + \mu_v^t \sigma_n^t}{\sigma_n^t + \sigma_v^t} - \frac{\mu_n^{t'} \sigma_v^{t'} + \mu_v^{t'} \sigma_n^{t'}}{\sigma_n^{t'} + \sigma_v^{t'}} \\ &= [(\mu_n^t - \mu_v^{t'})\sigma_v^t \sigma_n^{t'} + (\mu_v^t - \mu_n^{t'})\sigma_n^t \sigma_v^{t'} + (\mu_v^t - \mu_v^{t'})\sigma_n^t \sigma_n^{t'} \\ &\quad + (\mu_n^t - \mu_n^{t'})\sigma_v^t \sigma_v^{t'}]/[(\sigma_n^t + \sigma_v^t)(\sigma_n^{t'} + \sigma_v^{t'})] \\ &\geq [(\mu_v^t - \mu_v^{t'})\sigma_v^t \sigma_n^{t'} + (\mu_n^t - \mu_n^{t'})\sigma_v^t \sigma_n^{t'} + (\mu_v^t - \mu_v^{t'})\sigma_n^t \sigma_n^{t'} \\ &+ (\mu_n^t - \mu_n^{t'})\sigma_v^t \sigma_v^{t'}]/[(\sigma_n^t + \sigma_v^t)(\sigma_n^{t'} + \sigma_v^{t'})] \\ &\geq 0, \end{aligned} \tag{16}$$

According to (14), (15) and (16). We can easily conclude the targeted result that $S^*(t) \geq S^*(t')$. □

5.4 Hazard Ceiling: a Dynamic Perspective

As defined in Definition 5, hazard function is specifically denoted as $h(t) = -\frac{\mathrm{d}}{\mathrm{dt}} lnS(t) = -\frac{\mathrm{dS(t)}}{\mathrm{dt}} \cdot \frac{1}{S(t)}$, we can easily monitor the hazard function h(t) of a cascade when given its survival function S(t).

To detect the early pattern of outbreak cascades, many previous works usually ignore the underlying arrival process of retweets, instead, they only consider the relationship between the static size of cascade and a predefined threshold [6], [35], then determine whether the cascade is suffering a burst period. However, before the static size of a cascade accumulates to a certain threshold, its burst pattern can be exactly uncovered from dynamic information, such as the hazard function $h(t)$ in this problem.

Intuitively, we conclude that if at a certain time t_0, the hazard function $h(t)$ of a cascade suddenly rises above a *hazard ceiling* α, in other words, $h(t_0) > \alpha$, we deem that the burst period of this cascade begins.

However, instead of utilizing a fix threshold, we employ the baseline hazard function with a 5% hazard-tolerant interval as *hazard ceiling* (illustrated Fig.4c), since intuitively the characteristics of cascades may vary a lot during the diffusion process. In Fig.4c, the *hazard ceiling* is drawn in red dash line with a grey hazard-tolerant interval, and the red solid line and blue solid line respectively denote the hazard functions of a viral cascade and a non-viral cascade. We

can clearly conclude that the blue line never exceeds hazard ceiling α, and the red line exceeds α and its hazard-tolerant interval at t_{hazard}. Therefore, we deem that at t_hazard, this cascade goes viral and starts to burst. This can be helpful to make the prediction earlier, in other words, performing well in the early stage viral cascades detection task.

Here we propose two different hazard ceilings trying to catch the burst pattern of cascades. And we will explain the intuition behind them and do experiments to compare their effects in Section 6.

The **first** hazard ceiling utilizes the survival boundary function $S^*(t)$ and we call the *EPOC* model with this hazard ceiling as *EPOCs* in (17).

Definition 8 (Hazard Ceiling of EPOCs)**.** *we define the first hazard ceiling function as:*

$$h_{EPOCs}(t) = -\frac{\mathrm{dS^*(t)}}{\mathrm{dt}} \cdot \frac{1}{S^*(t)} \tag{17}$$

where $S^*(t)$ is the survival boundary function. Since the survival boundary originates from the idea that separating survival functions of viral and non-viral cascades in the average case, then $h_{EPOCs}(t)$, hazard function of $S^*(t)$, can represent the average boundary of the hazard function. So at any time t, if a cascade's hazard function exceeds the average case boundary $h_{EPOCs}(t)$, then this cascade meets a burst and is very likely to be viral.

The **second** hazard ceiling comes from the definition of viral cascades. As mentioned above, we use relative threshold ρ like the top 5% longest cascade's length. Then we can also use this proportion to obtain the hazard ceiling function and we call the *EPOC* model with this hazard ceiling as *EPOCr*.

The following is the formal definition: Given a set of k cascades' time series data $D = \{C_1, C_2, \cdots, C_k\}$ in the watching window with these cascades' feature matrix $F = \{X_1, X_2, \cdots, X_k\}$, we can get the survival function of these cascades by Cox's extended Model.

According to (2), we can get these cascades' hazard function. Denote i^{th} cascades hazard function as $h_i(t)$. Suppose we set the relative threshold ρ as the $(r \times k)^{th}$ longest cascade's length where r is a hyper parameter in the interval (0,1) and it is usually small.

At any time t, we denote $h_{threshold}(t)$ as the $(r \times k)^{th}$ largest value among all $h_i(t)(i = 1, 2, \cdots, k)$.

Definition 9 (Hazard Ceiling of EPOCr)**.** *Finally, we define the hazard ceiling function of EPOCr in (18).*

$$h_{EPOCr}(t) = h_{threshold}(t) \qquad \textit{at any time t} \tag{18}$$

The intuition behind the second hazard ceiling is that since we define the viral cascades with a ratio r, then we can also apply the ratio into the burst pattern detection. And the consistency of viral boundary ratio can be beneficial to keeping the accuracy while making the prediction earlier.

5.5 Incorporation of Static and Dynamic Techniques

In this subsection, we conclude our method and integrate survival boundary and hazard ceiling. The whole process of *EPOC* model is shown in Algorithm 1.

Algorithm 1: EPOC

Data: matrix of training and testing data D and D'
$D = \{C_1, C_2, \cdots, C_k\}$ where $C_i = \{t_1, t_2, \cdots, t_{n_i}\}$
$F = \{X_1, X_2, \cdots, X_k\}$ where $X_i = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_{n_i}}\}$
virality ratio r
Result: virality label vector $\boldsymbol{V}$ for D'
virality detected time vector $\boldsymbol{T}$ for D'

```
1  begin
2      Set the virality threshold ρ as the length of top
         (r * k)^th longest cascade among D and D'
3      Initialize real-label vector L for cascades of D and D'.
4      foreach cascades_i in D ∨ D' do
5          if n_i ≥ ρ then
6              L_i = 1
7          else
8              L_i = 0
9      Obtain time to event data D_s from D
10     Train Cox's model with D_s to obtain model
         parameter β by Maximum Likelihood Estimate
11     Obtain survival boundary S*(t) from D with β
12     Obtain the hazard ceiling function h_α(t)
13     foreach cascade_i' in D' do
14         Estimate the survival function S_i'(t) and
             hazard function h_i'(t) of cascade_i'
15         Initialize V_i' = 0 and T_i' = −1
16         for j = 1, 2, ··· , n_i' do
17             if S_i'(t_j) < S*(t_j) or h_i'(t_j) > h_α(t_j) then
18                 V_i' = 1
19                 T_i' = t_j
20                 break
21     return V and T
```

As we can see, *Line2* $\sim$ *Line9* is to label cascades. It is noteworthy that: we use full-length cascades during *labeling* training data. When it comes to *training* the model and *judging* the test data, we use cascades which are only former part of the origin cascades to simulate that in practice we need to predict the virality at the early stage. Then, *Line10* $\sim$ *Line11* obtains Cox Extended model's parameter β with *MLE* and *Newton method*. Next, *Line12* is to get the survival boundary according to Lemma 1. *Line13* obtains hazard ceiling function by the selected way, in our paper, either EPOCs or EPOCr. Finally, in *Line14* $\sim$ *Line23*, we use the survival boundary and hazard ceiling to get the label and the time when we predict that a cascade is viral for the test.

6 Experiment

In this section, we conduct comprehensive experiments to verify our model in early pattern detection of outbreak cascades. Firstly, we introduce some basic information about the data sets (Twitter and Weibo) and five comparative state-of-the-art baselines in details. Then we conduct our experiments as well as providing corresponding analysis.

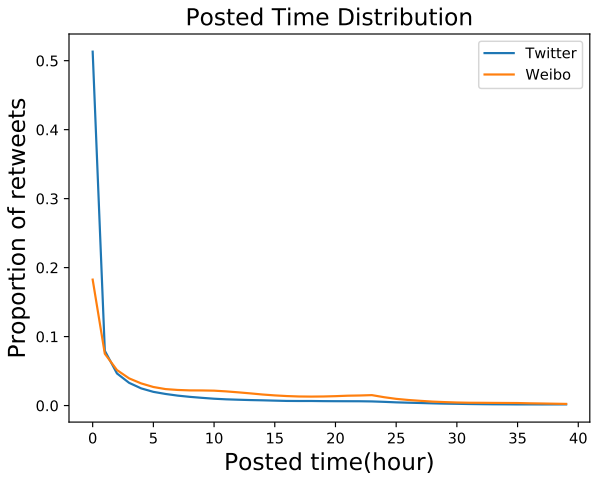


Fig. 5. Posted time distribution of Twitter and Weibo: The horizontal coordinate is the time span which equals to the retweets posted time minus the original tweet posted time and the vertical coordinate represents the proportion of retweets posted in this hour.

6.1 Data Sets

We implement our model *EPOC* on two large real-world data sets: Twitter and Weibo. Twitter is one of the most famous social platforms in the world with annually 5 billion users. We densely crawl the tweets that contains hashtags with Twitter search API. In our experiments, a cascade is considered to consist of all tweets with the same hashtag. Another large dataset Weibo is from an online resource[1]. However, different from Twitter, due to the sparsity of hashtags in Weibo, a cascade is defined by the diffusion of a single microblog. More detailed information of two data sets can be found in Table 2.

In addition, we also calculate the posted time distribution of Twitter and Weibo to understand the characteristics of datasets more deeply as in Fig.5. The horizontal coordinate is the time span which equals to the retweets posted time minus the original tweet posted time and the vertical coordinate represents the proportion of retweets posted in this hour.

As we can see in Fig.5, for both Twitter and Weibo dataset most of retweets are posted within 5 hours after the origin tweet posted. This observation tells us that if we can predict a cascade is viral with the data contains less than 5 hours length, then our prediction model is effective and has practical value, otherwise maybe useless. And we also can see: for Twitter, its retweets are concentrated more on early time than Weibo.

6.2 Baselines

From previous literatures, we select a variety of approaches from different perspectives to compare our *EPOC*: traditional machine learning methods, Bayesian methods, survival methods, and time series methods.

- *Linear Regression (LR)*: Linear regression is a simple and feasible way to characterize the relationship between variables and final result. In this paper, we divide the observation time into several time periods, then implement LR with L1 regularization based on different time periods, utilizing the observed information to predict whether or when a cascade goes viral. According to the regression result, we mainly consider two conditions. Given a viral cascade, if we can successfully predict it at time period $T_i(i = 1, 2, \cdots, n)$ but not before T_i, we deem that the prediction is correct, and the time-ahead can be calculated as the actual burst time minus T_i. On the contrary, if at any time period, we fail to predict that it is viral, we regard it as a wrong trial. The average time-ahead can be measured by the mean value of time-ahead in correct trials.
- *Support Vector Regression (SVR)*: As is widely used in various areas, SVR is a powerful regression model. We use SVR with Gaussian kernel as a baseline to predict whether a cascade will go viral or even burst in the near future. More detailed implementation of SVR is similar to the linear regression (LR).
- *PreWHether* [18] : From a Beyassian perspective, PreWHether is one of the pioneers in social content prediction, which utilizes three temporal features (sum, velocity, and acceleration) to infer the content ultimate popularity. It simulates that the features extracted from the previous frequency changes. In our experiments, we also use same time period manner to implement PreWHether.
- *SEISMIC* [10]: SEISMIC is a stochastic self-exciting point process based time series model, which takes individuals influence into consideration. It is a time-efficient model since it does not need any training. The self-exciting model thinks that every previous state can have an impact on the progress. Since the model itself is designed to predict the popularity of single tweets in social networks, we extend it to suit our goals of cascades burst pattern detection by the way that let the model predict every tweets final size and if the prediction exceeds the threshold, then we predict the cascade is viral.
- *SansNet* [8]: SansNet is a network-agnostic approach proposed based on survival theory as well in recent literature , which also regards the burst detection task as a judgement of viral and non-viral. This method shows its detection performance using only the time series information of a cascade only with the survival possibility model.

6.3 Evaluation Metric

In this subsection, we introduce 3 efficient and reasonable metrics to compare the evaluate and compare the results from different models or with different hyper-parameters.

Burst or Not: to detect the early pattern of outbreak cascades, we primarily divide this problem into two steps. Firstly, we detect whether a cascade will outbreak based on the observed information. Since large cascades are arguably more striking [28], in this classification task, we employ two special metrics: k-coverage and Cost.

K-coverage is a commonly used metric, which measures the proportion of detected outbreak cascades among the cascades that will break out actually. K-coverage mainly focuses on those cascades with a very large size. Specifically, it is calculated by $\frac{n}{k}(k \geq n)$, where k is the number of the largest cascades being concentrated on, and n denotes the number of cascades we successfully detect from the top-k viral cascades. Here in this work, n equals 50.

1. arnetminer.org/Influencelocality

TABLE 2
Datasets information

Data set	Range	Year	Size(GB)	# of cascades	# of tweets	Average length	Type
Twitter	Aug.13th - Sep.10th	2017	3.827	166,076	34,784,488	209	hashtag
Weibo	Sept.28th - Oct.29th	2012	1.426	300,000	42,380,016	141	microblog

Cost (more precisely called sensitive cost) is a targeted metric, which is selected to handle the problem of imbalanced data since viral cascades only take up a very small portion of all the cascades. If a viral cascade (like a rumor [1]) is classified to be non-viral, it will cost a lot when this cascade gets larger and causes a big trouble. On the contrary, if we misclassify a non-viral cascade, it does not matter and we just need some additional labor to check it and find it unimportant. To measure the different loss caused by different mistakes, we assign unequal cost to different mistakes based on their influence. The unequal cost equation is specified in (19):

$$Cost = \frac{FNR \times p \times Cost_{FN} + FPR \times (1-p) \times Cost_{FP}}{p \times Cost_{FN} + (1-p) \times Cost_{FP}} \tag{19}$$

where $FNR = \frac{FN}{FN+TP}$ is the false negative rate. $FPR = \frac{FP}{TN+FP}$ is the false positive rate, p is the proportion of viral cascades in all cascades, $Cost_{FN}$ and $Cost_{FP}$ are entries in cost matrix. The intuition behind the equation is that the numerator represents the expectation of the cost for the prediction result and the denominator represents the cost for the case of all wrong prediction which acts as normalization. We also specify the cost matrix in Table 3.

TABLE 3
Unequal-Cost Matrix

Real Class	Detected Class	
	Viral	**Non-viral**
Viral	$Cost_{TP} = 0$	$Cost_{FN} = 5$
Non-viral	$Cost_{FP} = 1$	$Cost_{TN} = 0$

Time Ahead (similar to EPA from [8]): As described above, we have two metrics to measure the correctness of the prediction results. Further, we want to figure out how early a model can detect the outbreak cascades. As [28] states, it is a pathological task to estimate the final size of a cascade if only given a short initial portion, since almost all cascades are small. Besides, comparing with getting the final size of a cascade, it is more meaningful and practical to detect how early a cascade will break out. Therefore, in this experiment of Twitter and Weibo, we only probe into the early pattern of outbreak cascades, and mainly focus on absolute time ahead (denoted as *ATA*), which is the interval between the predicted burst time tpredict and the actual burst time tactual. Specifically, during the experiments, if $t_{actual} \geq t_{predict}$, we record as $t_{actual} - t_{predict}$, and otherwise, 0. Also, we consider the relative time ahead (denoted as *RTA*), which is given by $\frac{t_{actual}-t_{predict}}{t_{actual}}$ or 0.

6.4 Experiment Setting

In this section, we will introduce settings of our model's parameters and the three different sets of experiments we have done in this paper.

The following are settings of our model implementation:

- Because large cascades are rare [28], in this paper, we noramally set threshold for viral and non-viral cascades to be 5% percentile in both Twitter and Weibo, where a cascade with larger size will be regarded as a viral one, and otherwise non-viral.
- As cascades are formed by large resharing activities and can potentially reach a large number of people [28], we only consider the cascades with a tweet count larger than 50 in Twitter and filter out the remains. As for Weibo, the out line is set to be 80.
- Since our dataset has long-time data of cascades, to simulate the task that we only have early time data of cascades to predict its virality, we just use full-length data to label the cascades and when it comes to training model and testing model, we only use start parts of cascades, in other words, we set an observable time window which are the same long for all cascades. And we set the observation window as 1 hours since most of the retweets are posted within 5 hours and we want to predict the virality of cascades as early as possible. Further, we also do experiments to analyze influence of the time window.
- For time-ahead metric, We implement it by: cutting each cascade into several small intervals and feeding these intervals to the model one by one. Once we predict a cascade is viral after feeding one new interval of this cascade into the model, then we stop feeding and call this interval is **detected time**. Finally, we obtain the time ahead by calculating the average of difference of real burst time minus detected time for those cascades which are both real viral and predicted viral.
- In the outset of our experiments, we randomly divide each data set into two parts, 80% of the cascades is employed as training data, and the remaining one-fifth as test data.
- To evaluate effects of the two different hazard ceilings we proposed, for our own models, we do experiments with 3 different hazard ceiling settings: the model with no hazard ceiling denoted as *EPOCn*, the model with the hazard ceiling mentioned in Definition 8 which is derived from survival boundary denoted as *EPOCs* and the model with hazard ceiling mentioned in Definition 9 which utilizes the threshold ratio denoted as EPOCr. With 5 other baselines, there are 8 models in total to be evaluated on all of our experiments.

The first set of experiments uses the above settings to compare those 8 models with 3 metrics: k-coverage, cost and time-ahead to evaluate their performances and find out their advantages and disadvantages.

Since our task is to predict the virality of cascades at early stage, the length of observing window is an important factor to be considered. Maybe some models do better than the other in predicting at the very early time while other

TABLE 4
Results of Different Models on Twitter and Weibo

		LR	SVR	PreWheter	SEISMIC	SansNet	EPOCn	EPOCs	EPOCr
Twitter	**k-coverage**	0.7652	0.5961	0.7492	0.5328	0.8035	0.8246	<u>0.8476</u>	0.8351
	Cost	0.1037	0.1013	0.0925	0.1617	0.0774	0.0752	0.0766	<u>0.0731</u>
	ATA(min)	441	254	472	165	549	565	<u>585</u>	578
	RTA	32.22%	24.62%	32.59%	15.02%	37.67%	40.44%	<u>43.58%</u>	42.47%
Weibo	**k-coverage**	0.6755	0.4983	0.6299	0.4636	0.7678	0.7692	<u>0.7803</u>	0.7783
	Cost	0.0903	0.1217	0.1280	0.1543	0.0937	0.0872	0.0889	<u>0.0867</u>
	ATA(min)	224	179	238	111	436	448	<u>467</u>	458
	RTA	25.33%	22.13%	27.93%	12.88%	33.75%	35.23%	<u>37.02%</u>	36.57%

models have a better performance when given a longer time window. So in the second set of experiments, we tune the time window length from 1 hour to 5 hours and evaluate their corresponding characteristics about the sensitivity to the observing window length.

Finally, we will analyze the influence of different threshold ratios to the prediction results because this ratio may vary according to their specific application area and we want to find out these models' robustness and specialty about threshold. So we tune the ratio of threshold from 5% to 25% in the third set of experiments and compare the results of different methods.

6.5 Model Comparison

In this section, we compare 8 models with 3 metrics: k-coverage, cost and time-ahead to evaluate their performances and find out their advantages and disadvantages.

The results are aggregated in Table 4 and the underlined numbers show the best results where *ATA* means Absolute Time Ahead, *RTA* means Relative Time Ahead, *EPOCn* is our model without hazard ceiling, *EPOCs* is our model with the hazard ceiling mentioned in Definition 8 which is derived from survival boundary and *EPOCr* is our model with hazard ceiling mentioned in Definition 9 which utilizes the threshold ratio.

As for Cost and k-coverage, all of our *EPOC* models perform relatively better than five baselines. LR also shows great performace in k-coverage on Weibo, and it works much better than SVR and SEISMIC, which means that the L1 regularization comes into effect. As a probabilistic model, PreWhether gives a slightly poor detection result due to the assumption that all the features are independent but that is fair since this modol is extremely time-efficient which does not need any training. Though less effective than *EPOC*, SansNet outperforms all the other baselines in this classification task, since SansNet only employs one feature from cascades. However, it is plausible to note that SansNet gives stable k-coverage and Cost results in both Twitter and Weibo, which indicates that survival perspective models are suitable in this scenario.

For *EPOCn*, *EPOCs* and *EPOCr*, we can see that hazard ceiling leads to better results because of its robust ability of estimating the instantaneous rates of the cascades growth. In detail, *EPOCs* perform better on k-coverage and worse on Cost than E*POCr*. This is mainly because:

- First, *EPOCs* obtains hazard ceiling from survival boundary by equation 4 and the survival boundary is intuitively a boundary between the average of viral and the average of non-viral cascades Since the survival boundary is a kind of average, it will not fluctuate wildly. As a result, its derivative will not be too large all the time, which means hazard ceiling of *EPOCs* will not be too large.
- In contrast, hazard ceiling of *EPOCr* is top r% largest $h(t)$. So it will be relatively large since all the time there may some cascades burst.
- Above all, it is more difficult for a cascade to pass through *EPOCr*'s hazard ceiling than *EPOCs*'. In other words, *EPOCr* is a more "conservative" model while *EPOCs* is more "aggressive", which can be observed from results in which *EPOCr* has highest k-coverage but relatively higher cost while *EPOCs* has lowest cost but relatively lower k-coverage. But both of them perform better than all the other models including *EPOCn*, which shows that hazard ceiling successfully catch the burst pattern and promote the prediction performance.

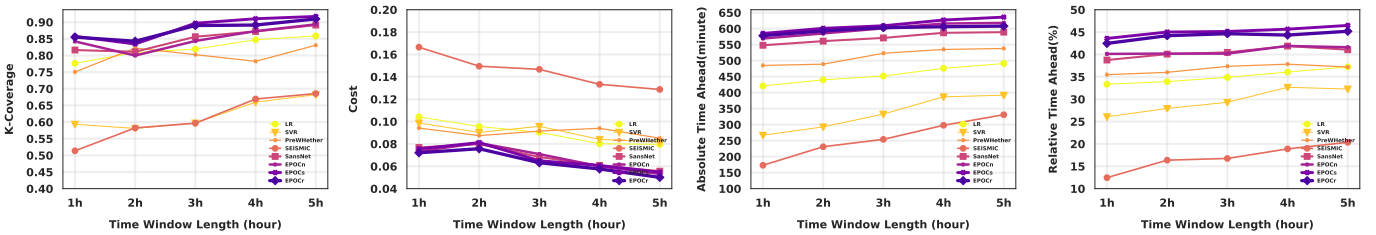
Besides, we can see that all the methods perform better on Twitter dataset than on Weibo. This is because as mentioned before, retweets on Twitter are more concentrated on early time than Weibo and we set the time window as 1 hour which is much shorter than 5 hours. Consequently, models on Twitter dataset have more data, in other words, more information and there is no wonder that all models have better results on Twitter dataset than Weibo.

6.6 Change of Observation Window Length

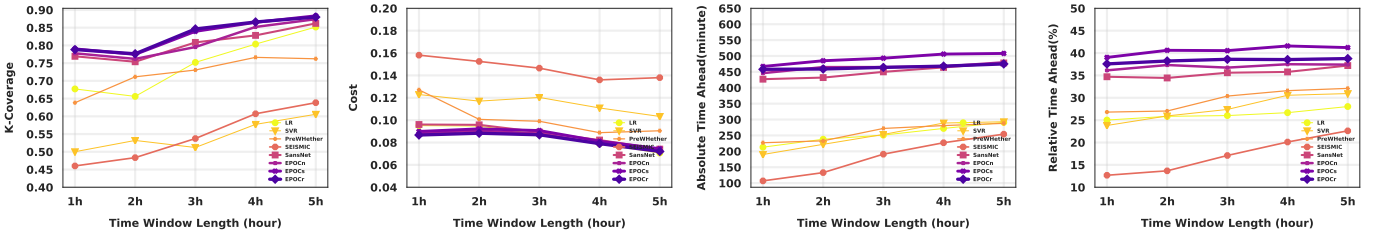
To explore the relation between length of observing window and the performances of these methods, we conduct experiments with 5 time periods from 1 hour to 5 hours on Twitter and Weibo dataset with 5 baselines, our *EPOCn*, *EPOCs* and *EPOCr* models.

The results by k-coverage, Cost and Time-Ahead on Twitter and Weibo are illustrated in Fig.6 and we can draw the following conclusions:

- As the time window becomes longer, almost all the models have better performance. And it makes sense because with more information accessible, prediction models can predict more accurately.
- In addition, we can see that the performance of SVR and SEISMIC grow fastest, LR and PreWhether slower, and all the survival models slowest. It is because SVR and SEISMIC are not so suitable for early stage prediction task, in other words, they perform not well when lack of information. But as the time window becomes longer,

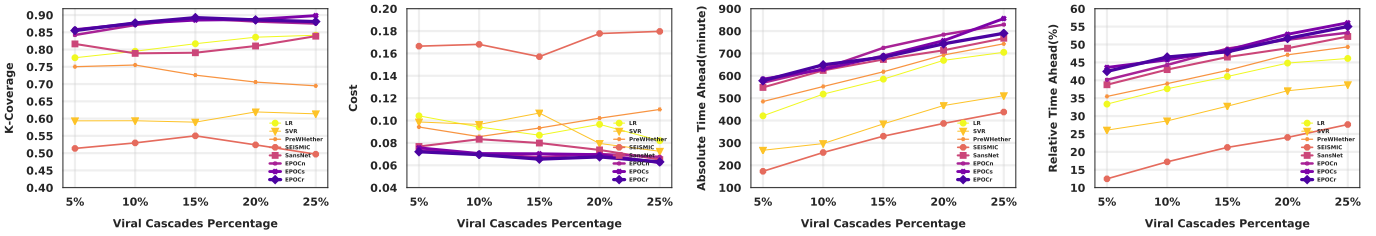


(a) K-coverage on Twitter (b) Cost on Twitter (c) Absolute Time Ahead on twitter (d) Relative Time Ahead on Twitter

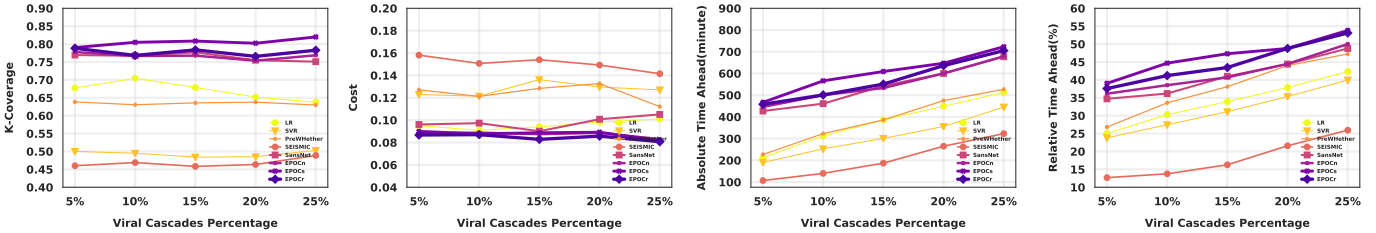


(e) K-coverage on Weibo (f) Cost on Weibo (g) Absolute Time Ahead on Weibo (h) Relative Time Ahead on Weibo

Fig. 6. Results with different length of time window



(a) K-coverage on Twitter (b) Cost on Twitter (c) Absolute Time Ahead on twitter (d) Relative Time Ahead on Twitter



(e) K-coverage on Weibo (f) Cost on Weibo (g) Absolute Time Ahead on Weibo (h) Relative Time Ahead on Weibo

Fig. 7. Results with different viral cascades ratio

they have more and more enough information to make prediction much better. On the other hand, survival models are designed for early stage prediction and perform relatively well with very few information. As a result, the size of their improvements are limited since they can predict at early time and additional information may not be so useful.

- All the models perform better on Twitter than on Weibo. This is because retweets on Twitter are concentrated more on early time then Weibo as illustrated in Fig.5. As a result, models can gain more information in the limited time window and achieve better performance.
- Our *EPOC* models outperform baselines all the time. As for *EPOCn*, *EPOCs* and *EPOCr*, *EPOCs* always has best k-coverage and time-ahead and *EPOCn* has best cost, which tallies with the discussion of Table 4.

6.7 Change of Viral Cascades Ratio

In this section, we will analyze the influence of different threshold ratios to the prediction results because this ratio may vary according to their specific application area and we want to find out these models' robustness and specialty about threshold. So we tune the ratio of threshold from 5% to 25%, in other words, the proportion of viral cascades from 5% to 25% and the results are shown in Fig.7. From it, we can conclude that:

- Our *EPOC* models perform better than other baselines at all the ratios which shows the robustness of our model.
- All the models' time-ahead metric score increases as the ratio of viral cascades increases. We think this phenomenon comes from the fact that: length of those relatively short cascades increases slowly and lower threshold means they can finally become viral but at very late time.

So, time-ahead metric's increase is mainly caused by those short-length newcomers.

- From the perspective of cost and k-coverage metric, different models have different best-performance ratios. For example, on Twitter, *EPOC* and SEISMIC perform best when ratio is 15%, SansNet, LR and SVR is 25% and PreWher is 10%. In addition, on different datasets, same model also has different best-performance ratios. For instance, *EPOC* have best k-coverage with ratio 15% on Twitter and with ratio 25% on Weibo. So we think people should set it carefully according to the specific applications. But all these models' performances show limited fluctuation with different ratios, which indicates that all the models are not so sensitive to this hyper-parameter.

7 Conclusion

In social networks, detecting whether and when a cascade will outbreak is a non-trivial but beneficial task. In this paper, we novelly employ survival theory, proposing a survival model *EPOC* to detect the early pattern of outbreak cascades. We extract both dynamic and static features from cascades and utilize Gaussian distributions to characterize their survival probabilities, then accompanied with hazard rate, we successfully detect the burst pattern of cascades at very early stage. Extensive experiment shows that our EPOC outperforms all those five state-of-the-art methods in this practical task.

As for future work, we have three directions to improve our *EPOC* models:

- Firstly, the hazard ceiling function of our model are derived from a relatively intuitive and experimental perspective. We think a hazard ceiling with more rough theoretical foundation may lead to better performance.
- Secondly, we will consider to add more influential and relevant features, such as the text content of the tweet, into out model so that we can have more accurate results with these additional information.
- Thirdly, we will try other methods in the survival theory to see if they can catch the survival characteristic of cascades better than Cox's Model.

Finally, we hope that our work will pave ways to richer and deeper understanding of cascades.

Acknowledgments

This work is supported by the National Key R&D Program of China 2018YFB1004703, the National Natural Science Foundation of China (61872238, 61672353), the Shanghai Science and Technology Fund (17510740200), Huawei Innovation Research Program (HO2018085286) and the State Key Laboratory of Air Traffic Management System and Technology (SKLATM20180X).

References

[1] A. Friggeri, L. A. Adamic, D. Eckles, and J. Cheng, "Rumor cascades," in *International AAAI Conference on Web and Social Media (ICWSM)*, 2014.

[2] J. Bai, L. Li, L. Lu, Y. Yang, and D. Zeng, "Real-time prediction of meme burst," in *IEEE International Conference on Intelligence and Security Informatics (ISI)*, 2017, pp. 167–169.

[3] Z.-Q. Jiang, W.-X. Zhou, D. Sornette, R. Woodard, K. Bastiaensen, and P. Cauwels, "Bubble diagnosis and prediction of the 2005–2007 and 2008–2009 chinese stock market bubbles," *Journal of Economic Behavior and Organization*, vol. 74, no. 3, pp. 149–162, 2010.

[4] D. Gruhl, R. Guha, R. Kumar, J. Novak, and A. Tomkins, "The predictive power of online chatter," in *ACM international Conference on Knowledge Discovery in Data Mining (SIGKDD)*, 2005, pp. 78–87.

[5] X. Ma, X. Gao, and G. Chen, "Beep: a bayesian perspective early stage event prediction model for online social networks," in *IEEE International Conference on Data Mining (ICDM)*, 2017, pp. 973–978.

[6] S. Wang, Z. Yan, X. Hu, S. Y. Philip, and Z. Li, "Burst time prediction in cascades." in *AAAI Conference on Artificial Intelligence (AAAI)*, 2015, pp. 325–331.

[7] Y. Matsubara, Y. Sakurai, B. A. Prakash, L. Li, and C. Faloutsos, "Rise and fall patterns of information diffusion: model and implications," in *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2012, pp. 6–14.

[8] K. Subbian, B. A. Prakash, and L. Adamic, "Detecting large reshare cascades in social networks," in *International Conference on World Wide Web (WWW)*, 2017, pp. 597–605.

[9] S. Gao, J. Ma, and Z. Chen, "Effective and effortless features for popularity prediction in microblogging network," in *International Conference on World Wide Web (WWW)*, 2014, pp. 269–270.

[10] Q. Zhao, M. A. Erdogdu, H. Y. He, A. Rajaraman, and J. Leskovec, "Seismic: A self-exciting point process model for predicting tweet popularity," in *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2015, pp. 1513–1522.

[11] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, "Everyone's an influencer: quantifying influence on twitter," in *ACM International Conference on Web Search and Data Mining (WSDM)*, 2011, pp. 65–74.

[12] S. Gao, J. Ma, and Z. Chen, "Modeling and predicting retweeting dynamics on microblogging platforms," in *ACM International Conference on Web Search and Data Mining (WSDM)*, 2015, pp. 107–116.

[13] Z.-H. Deng, X. Gong, F. Jiang, and I. W. Tsang, "Effectively predicting whether and when a topic will become prevalent in a social network," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2015, pp. 210–216.

[14] J. Pujara, H. Daumé III, and L. Getoor, "Using classifier cascades for scalable e-mail classification," in *ACM Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference (CEAS)*, 2011, pp. 55–63.

[15] E. M. Rogers, *Diffusion of innovations*. Simon and Schuster, 2010.

[16] A. Culotta, N. R. Kumar, and J. Cutler, "Predicting the demographics of twitter users from website traffic data." in *AAAI Conference on Artificial Intelligence (AAAI)*, 2015, pp. 72–78.

[17] S. Lin, X. Kong, and P. S. Yu, "Predicting trends in social networks via dynamic activeness model," in *ACM International Conference on Information and Knowledge Management (CIKM)*, 2013, pp. 1661–1666.

[18] Z.-H. Deng, X. Gong, F. Jiang, and I. W. Tsang, "Effectively predicting whether and when a topic will become prevalent in a social network," in *AAAI Conference on Artificial Intelligence (AAAI)*, 2015, pp. 210–216.

[19] T. Iwata, A. Shah, and Z. Ghahramani, "Discovering latent influence in online social activities via shared cascade poisson processes," in *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2013, pp. 266–274.

[20] E. Mansour, G. Tekli, P. Arnould, R. Chbeir, and Y. Cardinale, "Fsed: Feature-centric social event detection," in *International Conference on Database and Expert Systems Applications (DEXA)*, 2017, pp. 409–426.

[21] J. Zhang, B. Liu, J. Tang, T. Chen, and J. Li, "Social influence locality for modeling retweeting behaviors." in *International Joint Conferences on Artificial Intelligence (IJCAI)*, vol. 13, 2013, pp. 2761–2767.

[22] S. Petrovic, M. Osborne, and V. Lavrenko, "Rt to win! predicting message propagation in twitter." *International AAAI Conference on Web and Social Media (ICWSM)*, vol. 11, pp. 586–589, 2011.

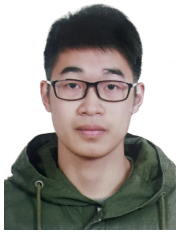
[23] P. Bao, H.-W. Shen, X. Jin, and X.-Q. Cheng, "Modeling and predicting popularity dynamics of microblogs using self-excited hawkes processes," in *International Conference on World Wide Web (WWW)*, 2015, pp. 9–10.

[24] H.-W. Shen, D. Wang, C. Song, and A.-L. Barabási, "Modeling and predicting popularity dynamics via reinforced poisson processes." in *AAAI Conference on Artificial Intelligence (AAAI)*, vol. 14, 2014, pp. 291–297.

[25] B. Ribeiro, M. X. Hoang, and A. K. Singh, "Beyond models: Forecasting complex network processes directly from data," in *International Conference on World Wide Web (WWW)*. International World Wide Web Conferences Steering Committee, 2015, pp. 885–895.
[26] S. Li, X. Gao, W. Bao, and G. Chen, "Fm-hawkes: A hawkes process based approach for modeling online activity correlations," in *ACM International Conference on Information and Knowledge Management (CIKM)*, 2017, pp. 1119–1128.
[27] L. Yu, P. Cui, F. Wang, C. Song, and S. Yang, "From micro to macro: Uncovering and predicting information cascading process with behavioral dynamics," *IEEE International Conference on Data Mining series (ICDM)*, 2015.
[28] J. Cheng, L. Adamic, P. A. Dow, J. M. Kleinberg, and J. Leskovec, "Can cascades be predicted?" in *International Conference on World Wide Web (WWW)*, 2014, pp. 925–936.
[29] L. Hong, O. Dan, and B. D. Davison, "Predicting popular messages in twitter," in *International Conference on World Wide Web (WWW)*, 2011, pp. 57–58.
[30] A. Kupavskii, L. Ostroumova, A. Umnov, S. Usachev, P. Serdyukov, G. Gusev, and A. Kustarev, "Prediction of retweet cascade size over time," in *ACM International Conference on Information and Knowledge Management (CIKM)*, 2012, pp. 2335–2338.
[31] Z. Feng, Y. Li, L. Jin, and L. Feng, "A cluster-based epidemic model for retweeting trend prediction on micro-blog," in *International Conference on Database and Expert Systems Applications (DEXA)*, 2015, pp. 558–573.
[32] D. Hunter, P. Smyth, D. Q. Vu, and A. U. Asuncion, "Dynamic egocentric models for citation networks," in *International Conference on Machine Learning (ICML)*, 2011, pp. 857–864.
[33] J. E. Jung, "Discovering social bursts by using link analytics on large-scale social networks," *Mobile Networks and Applications*, vol. 22, no. 4, pp. 625–633, 2017.
[34] G. Dong, W. Yang, F. Zhu, and W. Wang, "Discovering burst patterns of burst topic in twitter," *Computers and Electrical Engineering*, vol. 58, pp. 551–559, 2017.
[35] L. Hong, O. Dan, and B. D. Davison, "Predicting popular messages in twitter," in *International Conference on World Wide Web (WWW)*, 2011, pp. 57–58.
[36] P. Cui, S. Jin, L. Yu, F. Wang, W. Zhu, and S. Yang, "Cascading outbreak prediction in networks: a data-driven approach," in *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2013, pp. 901–909.
[37] O. Aalen, O. Borgan, and H. Gjessing, *Survival and event history analysis: a process point of view*. Springer Science and Business Media, 2008.
[38] D. R. Cox, "Regression models and life-tables," in *Breakthroughs in Statistics*. Springer, 1992, pp. 527–541.
[39] M. J. Fard, P. Wang, S. Chawla, and C. K. Reddy, "A bayesian perspective on early stage event prediction in longitudinal data," *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, vol. 28, no. 12, pp. 3126–3139, 2016.
[40] J. Yang and S. Counts, "Predicting the speed, scale, and range of information diffusion in twitter." *International AAAI Conference on Web and Social Media (ICWSM)*, vol. 10, no. 2010, pp. 355–358, 2010.
[41] M. Gomez-Rodriguez, J. Leskovec, and B. Schölkopf, "Modeling information propagation with survival theory," in *International Conference on Machine Learning (ICML)*, 2013, pp. 666–674.
[42] J. Anderson, L. Bernstein, and M. Pike, "Approximate confidence intervals for probabilities of survival and quantiles in life-table analysis," *Biometrics*, pp. 407–416, 1982.

Xiaofeng Gao received the B.S. degree in information and computational science from Nankai University, China, in 2004; the M.S. degree in operations research and control theory from Tsinghua University, China, in 2006; and the Ph.D. degree in computer science from The University of Texas at Dallas, USA, in 2010. She is currently an Associate Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, China. Her research interests include wireless communications, data engineering, and combinatorial optimizations. She has published more than 150 peer-reviewed papers in the related area, including well-archived international journals such as IEEE TC, TKDE, TMC, TPDS, JSAC, and also in well known conference proceedings such as SIGKDD, INFOCOM, ICDCS, etc. She has served on the editorial board of Discrete Mathematics, Algorithms and Applications, and as the PCs and peer reviewers for a number of international conferences and journals.

Xiaosong Jia is currently working towards a B.S. degree in Computer Science and Technology in the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University. He completed this work when he was working in the research group of Data Communication and Engineering. His research interests include data mining and machine learning on social networks.



Chaoqi Yang is currently a senior undergraduate in the School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University. He completed this work when he was working in the research group of Data Communication and Engineering. His research interests include data mining and machine learning on social networks and advertising.

Guihai Chen from Nanjing University in 1984, M.E. degree from Southeast University in 1987, and Ph.D. degree from the University of Hong Kong in 1997. He is a distinguished professor of Shanghai Jiao Tong University, China. He had been invited as a visiting professor by many universities including Kyushu Institute of Technology, Japan in 1998, University of Queensland, Australia in 2000, and Wayne State University, USA during September 2001 to August 2003. He has a wide range of research interests with focus on sensor networks, peer-to-peer computing, high-performance computer architecture and combinatorics. He has published more than 250 peer-reviewed papers, and more than 170 of them are in well-archived international journals such as IEEE Transactions on Parallel and Distributed Systems, Journal of Parallel and Distributed Computing, Wireless Networks, The Computer Journal, International Journal of Foundations of Computer Science, and Performance Evaluation, and also in well-known conference proceedings such as HPCA, MOBIHOC, INFOCOM, ICNP, ICPP, IPDPS and ICDCS.