CHASSIS: Conformity Meets Online Information Diffusion

[Technical Report]

Hui Li
School of Computer Sc. and Engg.,
Nanyang Technological University,
Singapore
HLI019@e.ntu.edu.sg

Hui Li
School of Cyber Engineering,
Xidian University, China
hli@xidian.edu.cn

Sourav S Bhowmick
School of Computer Sc. and Engg.,
Nanyang Technological University,
Singapore
assourav@ntu.edu.sg

ABSTRACT

Online information diffusion generates huge volumes of social activities (*e.g.,* tweets, retweets posts, comments, likes) among individuals. Existing information diffusion modeling techniques are oblivious to *conformity* of individuals during the diffusion process, a fundamental human trait according to social psychology theories. Intuitively, conformity captures the extent to which an individual complies with social norms or expectations. In this paper, we present a novel framework called CHASSIS to characterize online information diffusion by bridging classical information diffusion model with conformity from social psychology. To this end, we first *extend* "Hawkes Process", a well-known statistical technique utilized to model information diffusion, to quantitatively capture two flavors of conformity, *informational conformity* and *normative conformity*, hidden in activity sequences. Next, we present a novel *semi-parametric inference approach* to learn the proposed model. Experimental study with real-world datasets demonstrates the superiority of CHASSIS to state-of-the-art conformity-unaware information diffusion models.

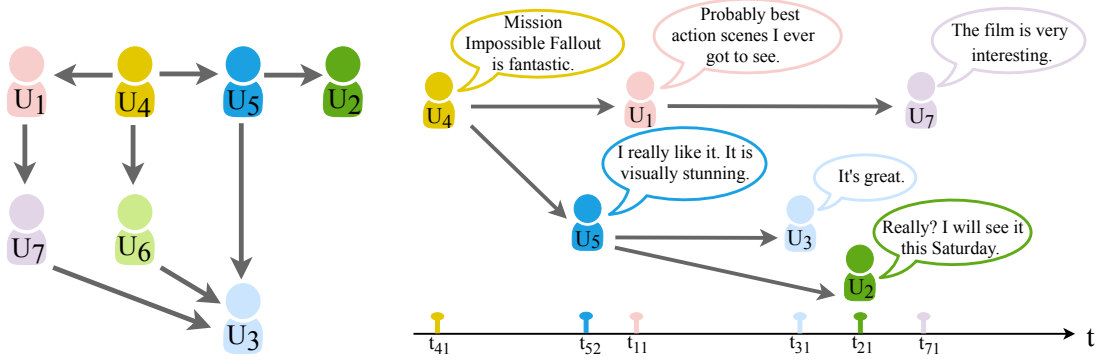
1 INTRODUCTION

Information diffusion is a process by which information and ideas spread over a network, creating a cascade. In particular, information diffusion in social networks continuously generates large-scale activities (*e.g.,* share, post, tweet, retweet, like, comment). Generally, these activities are generated in an asynchronous fashion since any individual can generate an activity at any time and there may not be any coordination or synchronization between two activities. Hence, such activities can be represented by asynchronous time-stamped sequences wherein each individual gives rise to a sequence of activities over time. In such sequences, there exist abundant *triggering relations* between activities that describe "which activity triggers which activity" [60, 61]. These relations are typically modeled as *diffusion trees* [33, 46]. For example, consider the social network in Figure 1(a) depicting follower-followee relationships. Figure 1(b) depicts a sequence of activities over time, involving some of the users, represented as a diffusion tree. Observe that an activity (*e.g.,* the post of U_4 at time t_{41}) may trigger a succeeding activity (*e.g.,* the comment of U_5 at time t_{52}) represented by a unidirectional link between them. Such unidirectional links between activities lead to diffusion trees. Hence, diffusion trees describe the information cascade (a.k.a informational cascade) generated by the information diffusion process. It is paramount to model this information diffusion process accurately as it underpins a variety of downstream applications such as influence maximization [4, 31], viral marketing [7, 44], rumour detection [41], user behaviour prediction [14].

Several studies have linked *conformity* [3, 5, 6], a fundamental and well-studied concept in social psychology, to the pivotal role it plays in the generation of information cascade [2, 9]. Intuitively, conformity refers to the inclination to align our attitudes and behaviors with those around us. There are two flavors of conformity, namely *informational conformity* and *normative conformity* [19]. The former occurs when people conform to peer views in an attempt to reach appropriate behaviors and attitudes due to lack of relevant knowledge. The latter occurs because of the desire to be accepted or that keep us from being isolated or rejected by others. For example, reconsider Figure 1. It is indeed possible that although U_3 is unaware of the movie "Mission Impossible Fallout", her response *"It's great"* is same as others because she chose to trust her friend U_5 (*i.e.,* informational conformity). On the other hand, suppose U_3 responds positively because she wants to please her friends even if she dislikes the movie. Then, this is an example of normative conformity.

Since conformity plays a fundamental role in how online users respond to social activities, it naturally influences the information diffusion process. Consequently, it is paramount for information diffusion models to incorporate it.

Example 1.1. Influence maximization (IM) aims to maximize the spread of information (or influence) in a network through activation of an initial set of *k seed* nodes [31]. The



(a) A social network (b) Diffusion process

Figure 1. Information diffusion.

dynamics of spread of information in a network is steered by an information diffusion model. The *Independent Cascade* (IC) is one of the most well-studied information diffusion models in this context [4]. In this model, in the first step a seed node is activated and subsequently at any time-step i, each newly activated node U_i gets one independent attempt to activate each of its outgoing neighbors U_j with a probability $p_{i,j}$, which is often set to $\frac{1}{\text{In}(j)}$ where $\text{In}(j)$ refers to the in-degree of U_j [4]. Observe that the IC model is conformity-unaware as the computation of $p_{i,j}$ disregards conformity of users.

Reconsider Figure 1(a). Based on the IC model, $p_{5,3} = \frac{1}{3}$ and $p_{5,2} = 1$. Thus, U_2 is more likely to be activated by U_5 than U_3. However, if we consider their responses in Figure 1(b), U_3 may exhibit higher degree of conformity to U_5 than U_2. Consequently, when conformity of individuals are taken into account, U_3 is more likely to be activated instead of U_2. Hence, a *conformity-aware* information diffusion model may potentially provide more accurate guidance to the IM problem. As several existing information diffusion models (*e.g.,* IC) are based only on the network structure, they fail to exploit information related to conformity of individuals. ∎

Despite the crucial role of conformity in online information diffusion, research in this arena is scarce [50, 59]. It is challenging to detect and quantify the two flavors of conformity from social activities. First, the private beliefs of individuals may not be exposed explicitly in the activities. For instance, U_3 may not explicitly mention in her post that she wants to please her friends or have not watched the movie. Hence, it may not be possible to determine whether an individual is conforming to another by simply searching for one's beliefs in the posts. Second, conformity of an individual may vary with the context. One may show high degree of conformity for one topic of discussion (*e.g.,* movies) but not another (*e.g.,* politics). Hence, any conformity computation technique needs to be *context-sensitive*. Third, the knowledge of topology of a social network is insufficient to address this problem as connectivities between individuals do not necessarily indicate manifestation of social activities among them. For instance, some followers may rarely or never interact with some of their followees, and some individuals may respond to some other unconnected individuals in online discussions. For example, in Figure 1, U_5 may respond to a comment by U_1 although they are not connected.

In this paper, we present a novel framework for information diffusion called ***C**onformity-aware **HA**wkes proce**SS**-based **I**nformation Diffu**S**ion* (CHASSIS) to characterize the underlying dynamics of diffusion in the presence of conformity. Specifically, we investigate how the aforementioned two flavors of conformity can be captured in individuals' interactions by exploiting diffusion trees constructed from the observed activity sequences.

Since social activities represent asynchronous time-stamped sequences, we deploy a well-known statistical technique called "Hawkes process" [27, 28], which is a type of point process* [54] that has been utilized recently to model information diffusion [56, 60]. Specifically, we *extend* classical Hawkes processes for information diffusion to capture time-varying conformity of individuals in our model (Section 4). We design a practical *semi-parametric approach* to learn the model components from observed data (Section 7) as well as *infer* the diffusion trees (Section 6) in an alternating fashion (*i.e.,* an instance of the Expectation-maximization method). To this end, as detailed in Section 3, we represent the diffusion trees by utilizing the *branching structure*, an equivalent representation of Hawkes processes, which isolates the events (*e.g.,* activities) in a Hawkes process into *immigrants* (*i.e.,* events that arrive independently) and *offsprings* (*i.e.,* events triggered by existing events). Then we utilize the parent-child pairs of events (*e.g.,* activities) in the branching structure (*i.e.,* diffusion trees), to quantify the two types

*Point processes are stochastic processes that are used to model events that occur at random intervals relative to the time or space axis, and provide the statistical language to describe the timing and properties of events.

of conformity in Section 5 and use them in our model. Extensive experiments with real-world datasets show superior performance of CHASSIS in modeling information diffusion compared to several state-of-the-art conformity-oblivious techniques. We also show that CHASSIS can be utilized to predict individuals' future behavior with considerable confidence, illustrating the powerful effects of an individual's inclination to align one's attitudes and behaviours with others during information diffusion.

In summary, this paper makes the following key contributions: (a) We propose a novel conformity-aware Hawkes process-based framework called CHASSIS to characterize online information diffusion. Our work *bridges the classical online information diffusion problem in data analytics with conformity from the domain of social psychology.* (b) We quantitatively capture two flavors of conformity, informational conformity and normative conformity, hidden in activity sequences by utilizing diffusion trees (*i.e.,* branching structure) constructed from the activity sequences. In this context, we propose a novel *diffusion tree inference* technique when explicit information about links between activities are unavailable to a downstream application. (c) We present a novel and efficient *semi-parametric inference approach* that leverages on the diffusion trees to learn the conformity-aware information diffusion model competently from observed data. (d) We conduct an experimental study with real social media datasets to demonstrate the superiority and effectiveness of CHASSIS and its ability to predict future behavior of individuals involved in information diffusion.

2 RELATED WORK

Conformity in online social networks. A rich line of work in social psychology [3, 5, 6, 10] has demonstrated the existence and importance of conformity in social interactions. However, there is scant research on investigating conformity in online social networks. The seminal work of Li *et al.* [35] studied the interplay between influence and conformity of each individual in online social networks by utilizing the positive and negative relationships between individuals. Subsequently, they modeled conformity in the context of IM problem [36]. Recently, [38] adopted group profiling in conformity-aware IM problem. Tang *et al.* [50] proposed a probabilistic factor graph model that predicts user behavior by exploiting the effect of conformity. The work in [59] assigns hidden roles to users and then learns the correlation between roles and conformity. None of these work model the interplay of informational and normative conformity, which is a more realistic way to capture the role conformity plays in social networks. Importantly, we focus on *inferring* the conformity-aware information diffusion model from the data, which is orthogonal to these efforts.

Opinion dynamic models [1, 8, 17] capture individuals' willingness to conform with the opinions of neighbors on a *certain* topic in social networks. These efforts fail to capture the information diffusion process.

Diffusion models. Information diffusion models study the hidden mechanism on how information spreads in a target social network. According to a previous study [26], they can be categorized into *predictive* and *explanatory* models.

Predictive models aim to uncover and predict how a specific diffusion process would unfold in a given network. These works consider the diffusion as a *discrete* random process happened among network nodes and can be further classified into *non-progressive* and *progressive* models. In the former model, a node affected by a piece of information cannot switch to unaffected status subsequently. This includes the independent cascade (IC) [44, 51] and linear threshold (LT) [21] models. They have been widely adopted to estimate and maximize the influence propagation within social networks [4]. IC/LT model has also been augmented with topic [13, 37], economic theory [7], and spatial-temporal features [34] in estimating the diffusion spread. In comparison, the *progressive model* (*e.g.,* SIR and SIS [29, 43] for virus propagation) allows an affected node to be unaffected again. All these predictive models are used for estimating the diffusion scope. They simplify the diffusion process to happen at discrete steps instead of continuous time.

Explanatory models are used to infer the underlying diffusion path in order to retrace and understand how a piece of information is propagated, and can benefit a series of applications including fake news detection [55], user behavior prediction [14], etc. For instance, [23] models the diffusion process as a spatially discrete network of *continuous*, conditionally independent temporal processes occurring at different rates. They presented NETRATE algorithm to infer pairwise transmission rates and the graph of diffusion. Recently, Hawkes process has been employed in modeling the information diffusion process. ADM4 [60] uses the mutually-exciting linear Hawkes model to capture the temporal patterns of user behaviors, and infer the social influence. MMEL [61] captures the temporal dynamics of the observed activities by utilizing multi-dimensional linear Hawkes processes, and learns the triggering kernels nonparametrically. Although these models are able to uncover the diffusion as a continuous temporal process, they fail to take into account conformity of individuals.

Lastly, there are also several efforts in the literature to predict the information cascade [14, 25, 57]. However, all these efforts are conformity-unaware.

Table 1. Key notations.

Notation	Definition
a_{ik}	the k^{th} activity by individual U_i
$N_i(t)$	the number of activities by U_i up to time t
t_{ik}	the occurrence time the activity a_{ik}
t_{ik}^-	time up to t_{ik} but not including t_{ik}^-
C_{ik}	the content of activity a_{ik}
Z_{ik}	parent activity of activity a_{ik}
(a_{jl}, a_{ik})	one parent-child pair of activities
$N_{ij}(t)$	collection of parent-child activity pairs by U_i, U_j
β_{ij}	the decay rate of previous interactions
$\gamma_{ij}^I(t)$	time-varying informational coefficient
$\alpha_{ij}^I(t)$	time-varying informational influence
$\gamma_{ij}^N(t)$	time-varying normative coefficient
$\alpha_{ij}^N(t)$	time-varying normative influence
$\mathbb{N}_i(t)$	number of offspring activities by U_i
LCA(a_{jl}, a_{ik})	the lowest common ancestor of a_{jl} and a_{ik}
p_{ik}	the polarity of activity a_{ik}
$p_{\text{LCA}(t_{jl}, t_{ik})}$	the polarity of activity LCA(a_{jl}, a_{ik})
$\mathbf{X}_i(t)$	the collection of activities by U_i up to time t
$\mathbf{X}_t$	the collection of activities up to time t
$\mathcal{H}_t$	the collection of activities before time t

3 BACKGROUND

In this section, we provide the necessary background knowledge to understand the paper. Key notations used in this paper are described in Table 1.

3.1 Hawkes Processes

Many applications may need to deal with timestamped events in continuous time. *Point process* is a principled framework to model such event data. Specifically, a point process on a time line is a random process for realization of the *event times* $t_1, t_2, \ldots$ falling along the line where t_i is the time of occurrence of the ith event (*e.g.,* a tweet, like). Point process can be equivalently represented as a counting process $N = \{N(t)|t \in [0, T]\}$ over the time interval $[0, T]$ where $N(t)$ records the number of events up to time t. Let $\mathcal{H}_t$ be the history of events before time t. Then dynamics of the point process could be characterized by a *conditional intensity function* $\lambda(t)$ as follows:

$$\lambda(t) = \lim_{\Delta t \to 0} \frac{\mathbb{E}[N(t + \Delta t) - N(t)|\mathcal{H}_t]}{\Delta t} \tag{3.1}$$

where two events coincide with probability 0, *i.e.,* $N(t + \Delta t) - N(t) \in \{0, 1\}$. Intuitively, the larger the intensity $\lambda(t)$, the greater the likelihood of observing an event in the time window $[t, t + \Delta t]$.

In some applications, the arrival of an event increases the likelihood of observing events in the near future. To model these applications, there exists a class of point processes in which the event arrival rate explicitly depends on past events. These processes are referred to as *self-exciting processes*. *Hawkes processes* [27] is the most well-known self exciting process and have been extensively used in many domains (*e.g.,* finance, seismology, social media).

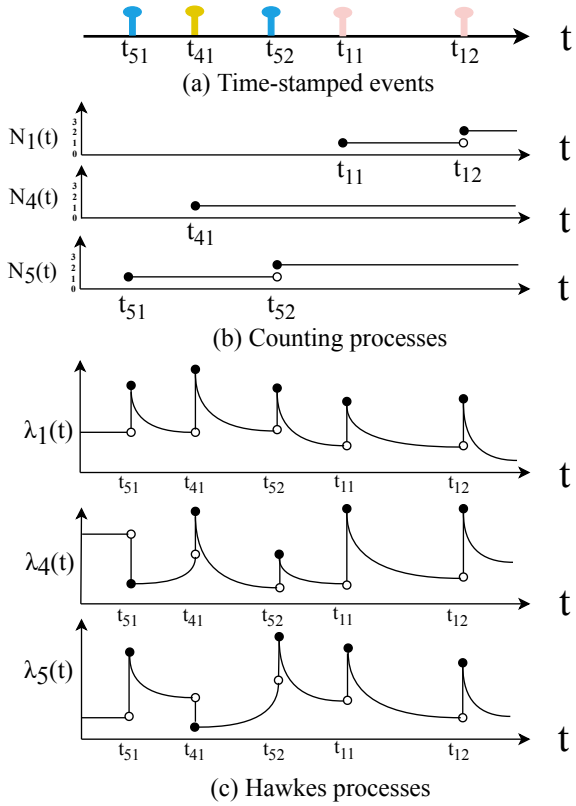


Figure 2. 3-dimensional Hawkes processes: (a) Five social activities during a time interval. (b) Counting process over time for each individual. $N(t)$ increases by one when an activity happens. (c) Intensity functions.

In this paper, we focus on *multi-dimensional Hawkes process* [60], which is defined by an M-dimensional point process where M Hawkes processes are *integrated* with each other. That is, it is an M-dimensional counting process where an arrival in one dimension can affect the arrival rates of all dimensions. *In information diffusion, each dimension i represents an individual U_i in a social network and an event represents a social activity.* Hence, each Hawkes process corresponds to an individual U_i and the influence between them is modeled by utilizing the *mutually-exciting* property of the M-dimensional Hawkes process. Formally, the intensity function of the ith dimension takes the following form [60]:

$$\lambda_i(t) = \mathcal{F}_i\left(\mu_i + \sum\nolimits_{j \in [M]} \sum\nolimits_{t_{jl} < t} \alpha_{ij} \phi_{ij}(t - t_{jl})\right) \tag{3.2}$$

Wherein the constant $\mu_i > 0$ is the *base intensity* of the ith Hawkes process, describing the arrival of events (*e.g.,* social activities) triggered by external sources. It is also referred to as *exogenous intensity*, and their arrivals are independent of the previous events. The strength of influence between

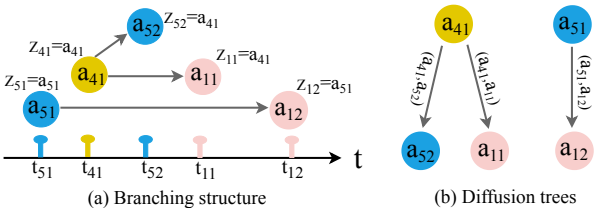


Figure 3. (a) The branching structure of the 3-dimensional Hawkes processes in Figure 2. (b) The corresponding diffusion trees.

dimensions (*i.e.,* individuals) is parameterized by a sparse *excitation matrix* $\boldsymbol{A} = [\alpha_{ij}]_{i,j\in[M]}$. In particular, the coefficient $\alpha_{ij} \geq 0$ captures the *mutually-exciting* property between the ith and jth processes. Larger value of α_{ij} indicates that events (activities) in the ith dimension are more likely to trigger an event in the jth dimension in the future. The *triggering kernel* $\phi_{ij}(t - t_{jl})$ quantifies the change in the rate of occurrence caused by the historical realization t_{jl}. The second item $\sum_{j\in[M]} \sum_{t_{jl}<t} \alpha_{ij}\phi_{ij}(t - t_{jl})$ is referred to as *endogenous intensity* and captures the mutually-exciting nature of the point processes. In our context, it captures the interactions between individuals in a social network – each event occurred to an individual U_j may increase (*i.e.,* mutual excitation) or decrease (*i.e.,* mutual inhibition) the arrival rate of occurrence in U_i by a certain amount which itself decays over time. Figure 2 depicts a 3-dimensional Hawkes processes involving U_1, U_4, and U_5 in Figure 1(a). Specifically, Figure 2(a) shows activity sequences at times t_{41}, t_{52}, and t_{11} in Figure 1(b) along with two additional activities by users U_5 and U_1 at times t_{51} and t_{12}, respectively. Figure 2(b) shows the corresponding counting process of each dimension. Figure 2(c) illustrates $\lambda_i(t)$ of the three individuals, provoking different changes due to these activities (*i.e.,* events). Observe that each occurred activity causes a jump (up or down) in the intensity function. Each jump is followed by a rapid decay guided by the kernel function.

Essentially, various combinations of kernel functions could recognize various temporal characteristics. When $\mathcal{F}_i(x) = x$, such processes are referred to as *linear* Hawkes processes [27] where the intensity is a linear accumulation of a series of kernel functions. Unfortunately, such linearity may not capture several real-world applications including information diffusion [42]. For example, a user may initially be extremely active on a particular topic in *Twitter*. However, her enthusiasm on that topic may subside eventually as she move on to a new topic. In another scenario, an individual's interest on a topic may be dampened (*i.e.,* inhibited) by posts from other users. Consequently, *nonlinear* Hawkes processes [11] are proposed to address this limitation. In this paper, we *integrate conformity with linear or nonlinear M-dimensional Hawkes processes for modeling information diffusion.*

3.2 Branching Structure

An equivalent view of the Hawkes process refers to the Poisson cluster process interpretation [28], which isolates the events in a Hawkes process into two categories: *immigrants* and *offsprings*. The offspring events are triggered by existing events in the process whereas the immigrants arrive independently and hence do not have an existing parent event. That is, we call an event an *immigrant* if it is generated due to the exogenous intensity $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_M)^T$ spontaneously, otherwise, it is an *offspring*. The offsprings are structured into *clusters*, associated with each immigrant event. This is referred to as the *branching structure* [27, 28]. It provides a way to capture the parent-child triggering relations between events as follows: (a) an immigrant event starts generating offsprings; (b) each offspring starts generating other offsprings immediately after birth.

To facilitate exposition in the context of social activity sequences, we denote a collection of activities (*i.e.,* events) as $\{a_{ik} = (t_{ik}, C_{ik})\}_{k=1}^{N_i(T)}$ in the time window $[0, T]$, where t_{ik} denotes the occurrence time of the k^{th} activity of an individual U_i, and C_{ik} records the activity's content. We introduce a set of auxiliary variables[†], denoted as $\{\{Z_{ik}\}_{k=1}^{N_i(t)}\}_{i=1}^{M}$, to represent the branching structure as following:

- $Z_{ik} = a_{ik}$ if activity a_{ik} is an immigrant; and
- $Z_{ik} = a_{jl}$ if the parent of activity a_{ik} is activity a_{jl}.

If activity a_{jl} triggers a_{ik}, (a_{jl}, a_{ik}) is referred to as a *parent-child* pair of activities (*i.e.,* event a_{ik} is an offspring of event a_{jl}). Given one offspring activity a_{ik}, we denote its parent activity as Z_{ik} accordingly, and the corresponding parent-child pair of activities as (Z_{ik}, a_{ik}).

Figure 3(a) depicts the branching structure representation of the Hawkes processes in Figure 2. Each circle represents an event (*i.e.,* activity) and a directed link represents the parent-child relationship between two events. For instance, $Z_{51} = a_{51}$ indicates that the activity a_{51} is an immigrant, and $Z_{12} = a_{51}$ denotes that the activity a_{51} generates a_{12}. Hence, (a_{51}, a_{12}) expresses a parent-child pair of events. Observe that *each connected component* in the branching structure represents a tree structure.

3.3 Diffusion Tree

A popular approach to represent a sequence of user activities over a time period is by using a collection of *diffusion trees* [33, 46], denoted by $\mathcal{D}$. A *diffusion tree*, $D_t = (V, E)$, consists of a set of user activities as its nodes V, and a set of unidirectional edges $E = \{(a_{ik}, a_{jl})\}$ denoting that the activity a_{ik} triggers the activity a_{jl} w.r.t the temporal precedence $t_{ik} < t_{jl}$. For example, in *Twitter*, the root node of D_t is an original tweet. If the original tweet triggers a series of

[†]W.l.o.g, we assume that an arbitrary event (activity) can be triggered by at most one event (activity). It is ubiquitous in practice (*e.g.,* Facebook, Twitter).

response, it generates a series of child activities (*e.g.,* retweet, comment, like), referred to as *first generation* descendants. Following this, first generation descendants subsequently generate their own child activities (*i.e., second generation* descendants), and so on. Figure 1(b) depicts a diffusion tree.

Observe the an original activity and its descendants in a diffusion tree represent an immigrant and its offsprings, respectively, in the branching structure. Hence, there is a direct correspondence between a set of diffusion trees and the branching structure of Hawkes processes. Each diffusion tree D_t is a connected component in the branching structure where a node and an edge in D_t are an event and a parent-child pair of events (*e.g.,* activities) in the latter, respectively. For example, the two diffusion trees in Figure 3(b) represent the branching structure in Figure 3(a). In the sequel, we shall use these two concepts interchangeably.

4 CONFORMITY-AWARE INFORMATION DIFFUSION MODEL

Equation 3.2 is used to model information diffusion by recent works [60, 61]. Specifically, its components are estimated from the observed social activities. Then, we can simulate the diffusion process beyond time T and predict various properties of the cascade. Observe that the strength of influence from an individual U_j to an individual U_i in Equation 3.2 (*i.e.,* α_{ij}) solely determines their degree of interaction in these classical Hawkes-based models. Intuitively, the stronger the influence α_{ij}, the more likely U_i responds to U_j during information diffusion. However, as remarked earlier, interactions between individuals are also likely to be impacted by conformity of users. That is, interactions between individuals not only depend on the strength of influence, but also on conformity of individuals. Hence, we need to augment classical information diffusion models to capture this phenomenon by incorporating informational conformity and normative conformity [19].

Although in some scenarios conformity may be purely informational or purely normative, in most cases these two occur concurrently [30]. The distinction between informational and normative conformity is at the functional level. The former is associated with accuracy and the search for information about reality whereas the latter is about social interactions [15]. Hence, one might conform for both normative and informational reasons at the same time [30]. Furthermore, contributions of these two types of conformity are likely to vary between different instances of conformity and between individuals [15]. Consequently, we decompose the time-varying *influence strength* $\alpha_{ij}(t)$ into two additive parts, *informational influence* $\alpha^I_{ij}(t)$ and *normative influence* $\alpha^N_{ij}(t)$, to quantify the presence of informational conformity and normative conformity, respectively. That is,

$$\alpha_{ij}(t) = \gamma^I_{ij}(t)\alpha^I_{ij}(t) + \gamma^N_{ij}(t)\alpha^N_{ij}(t) \tag{4.1}$$

In the above equation, the time-dependent *informational coefficient* $\gamma^I_{ij}(t)$ and *normative coefficient* $\gamma^N_{ij}(t)$ are parameterized to weigh informational conformity against normative conformity at time t. Observe that if $\alpha_{ij}(t) > 0$, then we know that conformity plays a role when U_j is influencing U_i. Substituting it into Eq. 3.2 gives us the model for conformity-aware Hawkes process-based information diffusion:

$$\lambda_i(t) = \mathcal{F}_i\left(\mu_i + \sum_{j\in[M]}\sum_{t_{jl}<t}(\gamma^I_{ij}(t)\alpha^I_{ij}(t) + \gamma^N_{ij}(t)\alpha^N_{ij}(t))\phi_{ij}(t-t_{jl})\right) \tag{4.2}$$

We elaborate on how to quantify $\alpha^I_{ij}(t)$ and $\alpha^N_{ij}(t)$ in the next section. In Section 7, we describe the inference of remaining components.

5 COMPUTATION OF CONFORMITY

In this section, we delineate how to quantify the two types of conformity using diffusion trees (*i.e.,* branching structure).

5.1 Informational Conformity

We often look to people around us who are better informed and more knowledgeable, and then use their opinions as a guide to our own behaviour and response. Such phenomenon (*i.e.,* the desire to be correct) not only occurs between friends but also individuals who have never known one another. This is known as informational conformity [12, 19]. Intuitively, informational conformity in social networks is not *symmetrical*. That is, informational conformity from an individual U_i to an individual U_j (*i.e.,* U_i conforms to U_j) may differ from that of U_j to U_i. According to social psychology theories [16, 47], the higher the influence of U_i, the higher the informational conformity of U_j to U_i. Following this, if U_j interacts with U_i frequently, then we should boost their *informational influence*. At the same time, during such interactions, if U_i almost always agrees with U_j, we can say that U_i is likely to conform to U_j. Consequently, we utilize the notions of *influence degree* (*i.e.,* measure of interaction frequency) and *context stance* (*i.e.,* opinion polarity *w.r.t* a topic) to quantify the pairwise informational conformity. Intuitively, the product of these two items describes how likely U_i's attitudes and behaviors are infected by another individual U_j in the presence of informational conformity. That is, informational influence from U_j to U_i (denoted as $\alpha^I_{ij}(t)$) is computed as follows:

$$\alpha^I_{ij}(t) = \Phi_{ij}(t) \times \Psi_{ij}(t) \tag{5.1}$$

wherein the first item $\Phi_{ij}(t)$ is referred to as *influence degree*, and the second item $\Psi_{ij}(t)$ aims to compute the *context stance*. Evidently, both of them are derived from the historical interactions between individuals. Put simply, the higher

the $\alpha_{ij}^I(t)$, the higher is the informational conformity and vice versa. We now elaborate on how these two factors are computed.

Influence Degree. Frequent interactions between individuals demonstrate their closeness, and lead to high pairwise influence degree [58]. Furthermore, such influence degree of one individual to another evolves over time. However, the effect of previous interactions may decrease with time, namely the time decaying effects [40]. For simplicity, we assume each response (*i.e.,* one offspring activity in the branching structure) provokes one interaction, followed by an exponential decay [49]. Hence, we measure the influence degree from individual U_j to individual U_i as:

$$\Phi_{ij}(t) = \frac{\sum_{k=1}^{N_i(t)} \mathbb{I}_{N_{ij}(t)}(Z_{ik}, t_{ik}) \exp\{-\beta_{ij}(t - t_{ik})\}}{\mathbb{N}_i(t)} \tag{5.2}$$

Wherein $N_{ij}(t)$ records the collection of parent-child activity pairs [‡], $\{(t_{jl}, t_{ik})\}$, up to time t. $\mathbb{I}_{N_{ij}(t)}(Z_{ik}, t_{ik})$ is an indicator function, which equals to one when $(Z_{ik}, t_{ik}) \in N_{ij}(t)$ and zero otherwise. We use $\boldsymbol{\beta} = \{\beta_{ij}\}$ to capture the decay rate of previous interactions between individuals. Different from $N_i(t)$, $\mathbb{N}_i(t)$ denotes the total number of offspring activities occurring to individual U_i until time t (*i.e.,* $\mathbb{N}_i(t) \leq N_i(t)$), which could be calculated by leveraging the diffusion trees of the activity collection $\mathbf{X}_t$. Obviously, the domain of influence degree from individual U_j to individual U_i is $[0, 1]$. Observe that $\Phi_{ij}(t)$ does not assume any connection between U_i and U_j (*i.e.,* U_i and U_j may or may not be friends/followers).

Context Stance. We glean insights on respondents' opinion polarity with respect to a topic in social interactions and apply stance detection [20] to obtain the dissemination of individuals' beliefs. Generally, such opinion polarity is often expressed in the form of discrete class labels, *e.g.,* positive or favor, negative or against, and neutral or none [20], either *explicitly* or *implicitly*. Explicit stances are direct expressions of opinion toward target concepts, such as "like" or "angry" given to a particular post and the corresponding polarity is 1 or 0, respectively. Implicit stances can be extracted from social media posts using *NLTK* (www.nltk.org), which is a popular sentiment analysis package[§].

Given each parent-child pair of activities (t_{jl}, t_{ik}) considered in $N_{ij}(t)$, we calculate the polarity p_{jl}, p_{ik} of activity t_{jl} and t_{ik}, and then append them into two vectors: $\overrightarrow{\boldsymbol{p}}_{\boldsymbol{j}}^{\boldsymbol{I}}(\boldsymbol{t}) = (p_{jl})_{a_{jl} \in \mathbf{X}_t}$ and $\overrightarrow{\boldsymbol{p}}_{\boldsymbol{i}}^{\boldsymbol{I}}(\boldsymbol{t}) = (p_{ik})_{a_{ik} \in \mathbf{X}_t}$, respectively. Next, we evaluate the Pearson correlation coefficient (Pcc) of the vectors, denoted as $\Psi_{ij}(t) = \text{Pcc}(\overrightarrow{\boldsymbol{p}}_{\boldsymbol{j}}^{\boldsymbol{I}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{p}}_{\boldsymbol{i}}^{\boldsymbol{I}}(\boldsymbol{t})) \in [-1, 1]$, to quantify the *context stance* over time. Intuitively, the higher the value of context stance, the higher is the informational conformity from individual U_i to individual U_j. The formal algorithm to compute informational conformity is given in Algorithm 1.

Consider the diffusion tree in Figure 1(b). We extract the opinion polarity of a_{11} and its response a_{71} (*i.e.,* (a_{11}, a_{71}) is a parent-child pair of activities). Suppose $p_{11} = 0.8, p_{71} = 0.9$. We append them into the vectors $\overrightarrow{\boldsymbol{p}}_{\boldsymbol{1}}^{\boldsymbol{I}}(\boldsymbol{t}) = [0.3, -0.7, 0.9]^T$ and $\overrightarrow{\boldsymbol{p}}_{\boldsymbol{7}}^{\boldsymbol{I}}(\boldsymbol{t}) = [-0.2, 0.5, 0.7]^T$, respectively, and then recalibrate the context stance by updating $\Psi_{71}(t) = \text{Pcc}(\overrightarrow{\boldsymbol{p}}_{\boldsymbol{1}}^{\boldsymbol{I}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{p}}_{\boldsymbol{7}}^{\boldsymbol{I}}(\boldsymbol{t}))$. Then we continue to scan the parent-child pairs of activities until time t, to capture the context stance.

Algorithm 1: InformationalConform Algorithm.

Input : $\mathbf{X}_t = \{\{a_{ik} = (t_{ik}, C_{ik})\}_{k=1}^{N_i(t)}\}_{i=1}^{M}$
Output: informational influence $\boldsymbol{\alpha}^{\boldsymbol{I}}(\boldsymbol{t})$

```
{D_t} ← DiffusionTreeConstruct(X_t);
/* Section 6                                          */
for i, j ∈ {1, ..., M} do
    Update ℕ_i(t), N_ij(t);
    /* according to {D_t}                             */
    for each pair (t_jl, t_ik) ∈ N_ij(t) do
        Update Φ_ij(t);
        /* according to Eq. 5.2                       */
        p⃗_j^N(t).append(p_jl);
        p⃗_i^N(t).append(p_ik);
        Update Ψ_ij(t) = Pcc(p⃗_j^N(t), p⃗_i^N(t));
        Update α_ij^I(t);
        /* according to Eq. 5.1                       */
    end
end
return α^I(t) = {α_ij^I(t)}_{i,j∈{1,...,M}}
```

Lemma 5.1. *The time complexity to compute informational complexity is $O(M^2 N_{max})$, where N_{max} is the maximum number of parent-child activity pairs in $\{N_{ij}\}_{i,j \in [M]}$ and M is the number of dimensions (*i.e., *individuals).*

5.2 Normative Conformity

Without loss of generality, a new post (*e.g.,* tweet) may lead to a chain of interactions (*e.g.,* retweet, comment, reply, like) in a social network. In practice, such an immigrant activity (*e.g.,* a tweet) generates its offspring activities (*e.g.,* a series of retweets, comments, replies, likes) possibly involving multiple individuals (*i.e.,* dimensions). In the sequel, we refer to an immigrant activity together with all its offspring activities as *one informational cascade*. Hence, based on Section 3.3, a diffusion tree D_t represents one informational cascade. For example, all activities in Figure 1(b) construct one informational cascade. Figure 4 depicts another example.

[‡] In the sequel, for simplicity, we sometimes use the occurrence time to denote one activity, *e.g.,* t_{jl} represents a_{jl}.
[§] The choice of sentiment analysis technique is orthogonal to our framework.

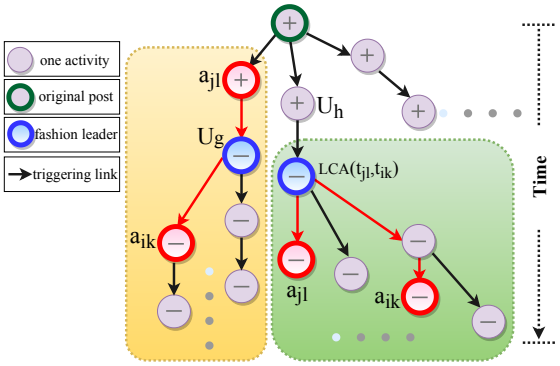


Figure 4. Diffusion tree of one informational cascade: "+" and "-" denote the opinion polarity (*i.e.,* adoption and rejection, respectively) of one activity *w.r.t* a topic.

An informational cascade occurs when it is optimal for an individual, having observed the actions of individuals ahead of him, to follow their behavior without regard to his own information [9]. In other words, a cascade can occur when people observe and follow "the crowd", even when the group consensus conflicts with their own private information [2]. Such phenomenon of following the crowd is known as normative conformity [12, 19].

Once one informational cascade starts around a particular topic, a few early individuals commit their actions (*i.e.,* adopt or reject) through a sequence of activities (*e.g.,* retweet, comment, reply, like), and then subsequent individuals may refer to them. Hence, triggering links between the preceding and following activities give us the opportunity to extract normative conformity by analysing the context stance hidden in the diffusion trees.

Specifically, the normative conformity of U_i to U_j depends on the *aggregated* adoptions of U_i to behaviors and attitudes of U_j. In reality, in one cascade, even though U_j's activity does not immediately precede U_i's, U_j's decision may convey information to U_i, and then U_i may act according to the information conveyed by the actions of preceding individuals (including U_j) [9]. Hence, in order to quantify the *normative influence* $\alpha_{ij}^N(t)$, we need to ensure the followings in the diffusion trees: (a) both individuals U_i and U_j are involved within one cascade; and (b) the corresponding activity a_{jl} happens before a_{ik}.

Furthermore, informational cascades can be fragile, with abrupt shifts or reversals in direction [9, 22]. Specifically, either small shocks (*i.e.,* when new information becomes available) can easily shift the behavior of many individuals, or higher-precision individuals can shift a cascade because they are more inclined to use their own information than those that precede them. For example, consider the activities of individuals highlighted in blue (also known as *fashion leader* [9]) in Figure 4. Observe that there is a shift in opinion polarity at this point, which is adopted by subsequent actions from individuals. Hence, the normative conformity of U_i to U_j may vary due to such sudden shifts.

We formulate the *normative influence* $\alpha_{ij}^N(t)$ as follows to capture how likely individual U_i's attitudes and behaviors are infected by another individual U_j:

$$\alpha_{ij}^N(t) = \text{Pcc}(\overrightarrow{\boldsymbol{p}}_j^{\boldsymbol{N}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{p}}_i^{\boldsymbol{N}}(\boldsymbol{t})) \tag{5.3}$$

In order to compute the context stance for this type of conformity (right side of the above equation), we consider the following two scenarios.

Scenario 1: Given an informational cascade, the two activities a_{jl} and a_{ik} *lie on the same path* of the diffusion tree in chronological order. No matter a_{jl} immediately precedes a_{ik} or not, U_j's action a_{jl} impacts U_i's response a_{ik} to some extent. In this case, we could directly capture their normative influence from the two activities in two steps: (a) append the polarity scores p_{jl}, p_{ik} into vectors $\overrightarrow{\boldsymbol{p}}_j^{\boldsymbol{N}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{p}}_i^{\boldsymbol{N}}(\boldsymbol{t})$, respectively; (b) recalibrate $\text{Pcc}(\overrightarrow{\boldsymbol{p}}_j^{\boldsymbol{N}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{p}}_i^{\boldsymbol{N}}(\boldsymbol{t}))$. For example, consider the yellow panel of Figure 4. Observe that U_j gives a positive response a_{jl} to the original post, and then U_g replied U_j with an opposiing view (*i.e.,* "-" denotes U_g's negative opinion polarity) due to some reason. Afterwards, U_i agrees with U_g. Obviously, U_g has a greater normative influence on U_i than U_j.

Scenario 2: Consider the two activities a_{jl} and a_{ik} located in the green panel of Figure 4. Even though they are triggered by different parent activities and are located in different paths of the diffusion tree, they are both impacted by the highlighted activity in blue (*i.e.,* the lowest common ancestor of a_{jl} and a_{ik}, denoting as $\text{LCA}(a_{jl}, a_{ik})$). Furthermore, if $\text{LCA}(a_{jl}, a_{ik})$ happens to be a fashion leader (*i.e.,* U_h gives positive response to the original post, afterwards, $\text{LCA}(a_{jl}, a_{ik})$ suddenly shifts his opinion), it definitely would have some effect on subsequent activities. Consequently, in such scenario, we quantify the normative conformity of U_i to U_j as follows. We first append the polarity pair p_{jl} (resp. p_{ik}) and $p_{\text{LCA}(t_{jl}, t_{ik})}$ into vectors $\overrightarrow{\boldsymbol{q}}_j^{\boldsymbol{N}}(\boldsymbol{t})$ (resp. $\overrightarrow{\boldsymbol{q}}_i^{\boldsymbol{N}}(\boldsymbol{t})$) and $\overrightarrow{\boldsymbol{q}}_{\text{LCA}_{ij}}^{\boldsymbol{N}}(\boldsymbol{t})$, and then recalculate their Pearson correlations coefficient $\text{Pcc}(\overrightarrow{\boldsymbol{q}}_j^{\boldsymbol{N}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{q}}_{\text{LCA}_{ij}}^{\boldsymbol{N}}(\boldsymbol{t}))$ (resp. $\text{Pcc}(\overrightarrow{\boldsymbol{q}}_i^{\boldsymbol{N}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{q}}_{\text{LCA}_{ij}}^{\boldsymbol{N}}(\boldsymbol{t}))$) before appending to $\overrightarrow{\boldsymbol{p}}_j^{\boldsymbol{N}}(\boldsymbol{t})$ (resp. $\overrightarrow{\boldsymbol{p}}_i^{\boldsymbol{N}}(\boldsymbol{t})$).

Scanning all information cascades satisfying the aforementioned conditions up to time t, we calculate the Pearson correlation coefficient of the vectors $(\text{Pcc}(\overrightarrow{\boldsymbol{p}}_j^{\boldsymbol{N}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{p}}_i^{\boldsymbol{N}}(\boldsymbol{t})))$ to quantify $\alpha_{ij}^N(t)$ between individuals in the presence of normative conformity. The formal algorithm to compute normative conformity is given in Algorithm 2.

Algorithm 2: NormativeConform Algorithm.

Input : $\mathbf{X}_t = \{\{a_{ik} = (t_{ik}, C_{ik})\}_{k=1}^{N_i(t)}\}_{i=1}^{M}$
Output: normative influence $\boldsymbol{\alpha}^{\boldsymbol{N}}(\boldsymbol{t})$

$\{D_t\} \leftarrow$ DiffusionTreeConstruct($\mathbf{X}_t$);
/* Section 6 */
for $i, j \in \{1, \ldots, M\}$ **do**
  Filter the collection of cascades C_{ij} satisfying the *two conditions*;
  /* according to $\{D_t\}$ */
  for *each cascade* $c \in C_{ij}$ **do**
    if a_{jl} *and* a_{ik} *lay on one path* **then**
      $\overrightarrow{\boldsymbol{p}}_j^{\boldsymbol{N}}(\boldsymbol{t})$.*append* (p_{jl});
      $\overrightarrow{\boldsymbol{p}}_i^{\boldsymbol{N}}(\boldsymbol{t})$.*append* (p_{ik});
      Update $\alpha_{ij}^N(t) = \text{Pcc}(\overrightarrow{\boldsymbol{p}}_j^{\boldsymbol{N}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{p}}_i^{\boldsymbol{N}}(\boldsymbol{t}))$;
    else
      $\overrightarrow{\boldsymbol{q}}_j^{\boldsymbol{N}}(\boldsymbol{t})$.*append*$(p_{jl})$;
      $\overrightarrow{\boldsymbol{q}}_i^{\boldsymbol{N}}(\boldsymbol{t})$.*append*$(p_{ik})$;
      $\overrightarrow{\boldsymbol{q}}_{\text{LCA}_{ij}}^{\boldsymbol{N}}(\boldsymbol{t})$.*append*$(p_{\text{LCA}(t_{jl}, t_{ik})})$;
      $\overrightarrow{\boldsymbol{p}}_j^{\boldsymbol{N}}(\boldsymbol{t})$.*append* $\left(\text{Pcc}(\overrightarrow{\boldsymbol{q}}_j^{\boldsymbol{N}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{q}}_{\text{LCA}_{ij}}^{\boldsymbol{N}}(\boldsymbol{t}))\right)$;
      $\overrightarrow{\boldsymbol{p}}_i^{\boldsymbol{N}}(\boldsymbol{t})$.*append* $\left(\text{Pcc}(\overrightarrow{\boldsymbol{q}}_i^{\boldsymbol{N}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{q}}_{\text{LCA}_{ij}}^{\boldsymbol{N}}(\boldsymbol{t}))\right)$;
      Update $\alpha_{ij}^N(t) = \text{Pcc}(\overrightarrow{\boldsymbol{p}}_j^{\boldsymbol{N}}(\boldsymbol{t}), \overrightarrow{\boldsymbol{p}}_i^{\boldsymbol{N}}(\boldsymbol{t}))$;
    end
  end
end
return $\boldsymbol{\alpha}^{\boldsymbol{N}}(\boldsymbol{t}) = \{\alpha_{ij}^N(t)\}_{i,j \in \{1,\ldots,M\}}$

Note the difference in the computation of context stance for normative conformity compared to informational conformity. For the latter, an individual may refer to surrounding people who are better informed and more knowledgeable, and then use their opinion as a guide for his/her own behaviours. Hence, computation of the context stance for informational conformity focuses on the parent-child activity pairs (*i.e.,* U_j precedes U_i immediately). For the former, an individual follows the behaviour of the preceding individuals during an informational cascade. Consequently, context stance of U_i and U_j is computed by considering the aggregated activities of U_i to the activities of U_j even though U_j's activity may not immediately precede U_i's activity (*i.e.,* they are not parent-child activity pairs).

Lemma 5.2. *The normative conformity computation requires $O(M^2m^2n)$ time, where n is the number of informational cascades, $m = \max |C_{ij}| \ll M$ is the maximum number of activities in a single cascade and M is the number of individuals.*

6 CONSTRUCTION OF DIFFUSION TREES

In the preceding section, the informational and normative influence (*i.e.,* $\alpha_{ij}^I(t)$, $\alpha_{ij}^N(t)$) are computed by utilizing the diffusion trees. In this section, we elaborate on how the diffusion trees are construction.

Connectivity-aware construction. If an online social network explicitly exposes connectivity information (*i.e.,* parent-child link) of activity sequences to an application then it is straightforward to construct the diffusion trees. That is, if a collection of social activities explicitly contain information of which activity responds to which activity (*e.g., reply_id*), we could establish the parent-child pairs of activities and construct the diffusion trees (*i.e.,* branching structure) accordingly. Activities with no parents form the immigrants and those with parents form the offsprings.

Diffusion tree inference. The construction of diffusion trees becomes challenging when parent-child link information is unavailable from social activities exposed to an application (*e.g.,* links in Figure 3 are missing). For example, the *Twitter* API returns the following fields: *tweet_id*, *created_time*, *text*, and *user_id*[¶]. That is, it does not provide connectivity information (*e.g., reply_id*) of the activities. Hence, we need to *infer* the latent diffusion trees (*i.e.,* the branching structure).

Branching structure or diffusion tree inference for information diffusion has been addressed in several prior work [60, 61]. Unfortunately, these existing methods are only suitable for *linear* Hawkes processes. As remarked earlier, in our problem setting Hawkes processes can be linear or nonlinear. In particular, the nonlinear combinations of exogenous and endogenous intensities (*i.e.,* various forms of $\mathcal{F}_i$ in Eq. 3.2) make such decomposition untenable in this scenario. Hence, it is desirable to devise an inference strategy that can handle both types of Hawkes processes by relaxing the requirement of linearity.

Furthermore, observe that this problem is orthogonal to the classical *network inference problem* [24, 39], which focuses on predicting links (connections) between individuals (*e.g.,* Figure 1(a)). In contrast, as illustrated in Figures 1(b) and 3, our goal is to infer the links between social activities to reveal the information cascade during information diffusion. Hence, techniques designed for network inference cannot be adopted to address this problem.

We propose an expectation-maximization (EM) iterative learning scheme to infer the diffusion trees. To initialize the EM procedure, we firstly sample the auxiliary variables $\{\{Z_{ik}\}_{k=1}^{N_i(t)}\}_{i=1}^{M}$. Afterwards, we update the probability of branching structure (*i.e.,* infer the diffusion trees) in the E-step given the CHASSIS model learned from the previous iteration. Thus, the inference procedure of CHASSIS can be embedded into the M-step naturally. In this section, we elaborate on the inference of diffusion trees. We defer the details of the inference procedure of CHASSIS in Section 7.

[¶]http://socialmedia-class.org/twittertutorial.html

The intuition behind our inference strategy is as follow. The greater the influence of the preceding activity to the following activity, the more likely there is a triggering link between them (*i.e.,* they are a parent-child pair of activities). From this perspective, we first deduce the *Papangelous conditional intensity* [53] of Hawkes processes to weigh the extent to which removing one activity will affect the subsequent activities in chronological order. Then, we utilize it to reflect the probability that a preceding activity a_{ik} triggers on a succeeding activity a_{jl}. After that, we obtain the parent-child pairs of activities probabilistically. Observe that our strategy of exploiting Papangelous conditional intensity to calculate the influence weight between activities is novel, as existing strategies compute conditional intensity conditioned *only* on historical activities not including subsequent ones. We now elaborate on these steps.

Firstly, following [53], we deduce the Papangelous conditional intensity of Hawkes processes, denoted as λ^p, which describes the probability of finding one point (*i.e.,* one time-stamped activity) at one particular time conditional on the remainder of the process [53]. Equivalently, Papangelous conditional intensity could cover the impact from both anterior and posterior activities. Hence,

$$\lambda^p(t_{jl}|\{\{t_{ik}\}_{k=1}^{N_i(t)}\}_{i=1}^{M}) = \frac{f(\{\{t_{ik}\}_{k=1}^{N_i(t)}\}_{i=1}^{M})}{f(\{\{t_{ik}\}_{k=1}^{N_i(t)}\}_{i=1}^{M} \setminus t_{jl})} \tag{6.1}$$

characterizes the density of particular sequences of activities. $\{\{t_{ik}\}_{k=1}^{N_i(t)}\}_{i=1}^{M} \setminus t_{jl}\}$ indicates the remaining activities after removing t_{jl} from the collection $\{\{t_{ik}\}_{k=1}^{N_i(t)}\}_{i=1}^{M}$. Note that the numerator on the right side is constant once given the collection of activities, and the value of the denominator quantifies how much of an effect removing activity t_{jl} will have on the reminder of the processes. Consequently, by removing each activity before t_{ik} respectively, we can measure the corresponding influence to activity a_{ik}. The higher the reciprocal of such Papangelous conditional intensity, denoted as $\frac{1}{\lambda^p(t_{jl}|\,(\{t\}_{t:t\le t_{ik}}\setminus t_{jl})}$, the more likely that a_{jl} triggers a_{ik}. Accordingly, we weigh the following probability:

$$\mathcal{P}(Z_{ik} = a_{jl}) = \frac{1}{\lambda^p(t_{jl}|\,(\{t\}_{t:t\le t_{ik}} \setminus t_{jl})} \tag{6.2}$$

$$\mathcal{P}(Z_{ik} = a_{ik}) = \frac{1}{\lambda^p(t_{ik}|\,(\{t\}_{t:t\le t_{ik}} \setminus t_{ik})} \tag{6.3}$$

wherein $\{t\}_{t:t\le t_{ik}}$ is the collection of chronologically ordered activities from M individuals up to time t_{ik} including t_{ik} and $t_{jl} < t_{ik}$. Observe that the above posterior for an arbitrary activity a_{ik} follows the Multinomial posterior distribution. Given the activity a_{ik}, we calculate the probability of each preceding activity (including a_{ik} itself), denoted as a_{jl}, as its parent activity Z_{ik}, and then apply *soft-update-rule* to determine its parent activity if any. For instance, if $\max\{\mathcal{P}(Z_{ik} = a_{jl})\} = \mathcal{P}(Z_{ik} = a_{hm})$, we conclude that activity a_{hm} triggers activity a_{ik}, and (a_{hm}, a_{ik}) is a parent-child pair. In particular, if $\max\{\mathcal{P}(Z_{ik} = a_{jl})\} = \mathcal{P}(Z_{ik} = a_{ik})$, we represent that activity a_{ik} as an immigrant.

Reconsider Figure 1(b). In order to exploit which activity triggers a_{71}, we calculate $\mathcal{P}(Z_{71} = a_{ij})$ according to Eq. 6.2 wherein a_{ij} equals to each activity happened up to time t_{71} (including t_{71}). Finally, while obtaining $\max\{\mathcal{P}(Z_{71} = a_{ij})\} = \mathcal{P}(Z_{71} = a_{11})$, we draw one link from a_{11} to a_{71}, and conclude that the parent activity of a_{71} is a_{11}. Given an arbitrary activity a_{ik}, we explore its parent activity Z_{ik} if any, then splice them together. Consequently, the set of auxiliary variables $\{Z_{ik}\}$ represents the branching structure (collection of diffusion trees).

In order to calculate the right side of Eq. 6.1, we set $f^*(t) = f(t|\mathcal{H}_t)$ to be the conditional density function of the occurrence time of the next activity given the history [54]. Hence, the distribution of all activities up to time t could be derived by the joint density,

$$f(\{\{t_{ik}\}_{k=1}^{N_i(t)}\}_{i=1}^{M}) = \prod_{i=1}^{M}\prod_{k=1}^{N_i(t)} f(t_{ik}|\mathcal{H}_{t_{ik}}^{-}) = \prod_{i=1}^{M}\prod_{k=1}^{N_i(t)} f^*(t_{ik}) \tag{6.4}$$

Analogically, the denominator equals:

$$f(\{\{t_{ik}\}_{k=1}^{N_i(t)}\}_{i=1}^{M} \setminus t_{jl}) = \frac{1}{f^*(t_{jl})}\prod_{i=1}^{M}\prod_{k=1}^{N_i(t)} f^*(t_{ik}) \tag{6.5}$$

Supposing $t_{in} < t < t_{i(n+1)}$, we could write the conditional intensity function of the i-th dimensional Hawkes process in terms of the conditional density function $f^*(t)$ and its corresponding cumulative distribution function $F^*(t)$ [54] as follows:

$$\begin{aligned}
\lambda_i(t)dt &= \mathcal{P}\big(t_{i(n+1)} \in [t, t+dt]|\mathcal{H}_{t^-}\big) \qquad (6.6)\\
&= \mathcal{P}\big(t_{i(n+1)} \in [t, t+dt]|t_{i(n+1)} \notin [t_{in}, t], \mathcal{H}_{t_{in}}\big)\\
&= \frac{\mathcal{P}\big(t_{i(n+1)} \in [t, t+dt], t_{i(n+1)} \notin [t_{in}, t]|\mathcal{H}_{t_{in}}\big)}{\mathcal{P}\big(t_{i(n+1)} \notin [t_{in}, t]|\mathcal{H}_{t_{in}}\big)}\\
&= \frac{\mathcal{P}\big(t_{i(n+1)} \in [t, t+dt]|\mathcal{H}_{t_{in}}\big)}{\mathcal{P}\big(t_{i(n+1)} \notin [t_{in}, t]|\mathcal{H}_{t_{in}}\big)}\\
&= \frac{f(t|\mathcal{H}_{t_{in}})dt}{1-F(t|\mathcal{H}_{t_{in}})} = \frac{f(t|\mathcal{H}_t)dt}{1-F(t|\mathcal{H}_t)} = \frac{f^*(t)}{1-F^*(t)}
\end{aligned}$$

Consider an infinitesimal interval dt around t. Then $f^*(t)dt$ corresponds to the probability that there is an activity in dt, and $1 - F^*(t)$ corresponds to the probability of no new activities before time t. Hence, according to Eq. 6.6, the conditional density function becomes [52],

$$f^*(t) = e^{-\int_0^t \lambda_i(s)ds} \cdot \prod_{k=1}^{N_i(t)} \lambda_i(t_{ik}) \tag{6.7}$$

By substituting Eq. 4.2 into Eq. 6.7, 6.4, 6 and Eq. 6.1, we obtain the Papangelous conditional intensity of our proposed conformity-aware Hawkes processes.

Algorithm 3: DiffusionTreeConstruct.

Input : $\mathbf{X}_t = \{\{a_{ik} = (t_{ik}, C_{ik})\}_{k=1}^{N_i(t)}\}_{i=1}^{M}$
Output: $\{\{Z_{ik}\}_{k=1}^{N_i(t)}\}_{i=1}^{M}$

```
while not converged do
    Update the intensity in Eq. 4.2; /* Section 7 */
    for each activity a_ik ∈ X do
        Calculate P(Z_ik = t_ik); /* Eq. 6.3 */
        P_ik = P(Z_ik = a_ik);
        for each proceding activity a_jl before t_ik do
            Calculate P(Z_ik = t_jl); /* Eq. 6.2 */
            Update P_ik = max{P(Z_ik = a_jl), P_ik}
        end
        Update Z_ik according to P_ik;
    end
end
return {{Z_ik}_{k=1}^{N_i(t)}}_{i=1}^{M}
```

The overall approach for inferring the diffusion trees is as follows. After each iteration for inferencing CHASSIS in M-step (Section 7), we update the intensity $\{\lambda_i(t)\}_{i\in[M]}$ by substituting the estimated parameters and triggering kernels. Given an activity, we evaluate the effect from removing each activity before it by deducing the corresponding Papangelous conditional intensity from the updated intensity. Subsequently, we find the parent activity for each activity (one activity has at most one parent activity) using the aforementioned approach to construct the diffusion trees. The formal algorithm is given in Algorithm 3.

LEMMA 6.1. *The time complexity to infer the diffusion trees is $O(N_{iter}n^2)$, where n is the total number of activities.*

7 INFERENCING CHASSIS MODEL

Once the diffusion trees are updated, we optimize the CHASSIS model to best explain the information diffusion process. Consequently, in this section we propose a novel semi-parametric inference algorithm regardless of whether Hawkes processes are linear or nonlinear, wherein exogenous intensity $\{\mu_i\}_{i\in[M]}$, decay rate of previous interactions $\{\beta_{ij}\}_{i,j\in[M]}$, informational and normative coefficients $\{(\gamma_{ij}^I(t), \gamma_{ij}^N(t))\}_{i,j\in[M]}$ are learned from the observed activity sequences, while the triggering kernel functions $\{\phi_{ij}(t)\}_{i,j\in[M]}$ are estimated nonparametrically via Fourier transform without prior domain knowledge.

7.1 Parametric Inference

We denote the set of parameters $\{\mu_i, \gamma_{ij}^I(t), \beta_{ij}, \gamma_{ij}^N(t)\}_{i,j\in[M]}$ as Θ. We can estimate them by maximizing the likelihood over the observed data (*i.e.,* maximize the likelihood function). Given the social activity collection $\mathbf{X}_t$ over the time interval $(0, t]$, the log-likelihood[‖] of conformity-aware Hawkes processes associated with the conditional intensity in Eq. 4.2 is in fact the summation of that over all dimensions, each of which can be interpreted as follows: the sum of the log-intensities of activities that happened, minus an integral of the total intensities over the observation interval $(0, t]$ [60],

$$\ln \mathcal{L}_i(\Theta|\mathbf{X}_t) = \sum\nolimits_{k=1}^{N_i(t)} \ln \lambda_i(t_{ik}) - \int_0^t \lambda_i(s)ds \tag{7.1}$$

However, $\int_0^t \lambda_i(s)ds$ is not always directly computable *w.r.t* various intensity functions $\mathcal{F}_i(\cdot)$. We propose a modified flexible-size Euler integration method[**] to calculate $\int_0^t \lambda_i(s)ds$ in an iterative manner.

THEOREM 7.1. *Let $\Lambda_i(t) = \int_0^t \lambda_i(s)ds$. Given an accuracy bound ξ, within the time interval $(0, t]$, taking I_m steps with the Euler method using step size $h_m = \frac{t}{I_m}$, the m^{th} iteration yields the following approximation*

$$\Lambda_i^m(t) = h_m\big(\mu_i + \lambda_i(t_1) + \ldots + \lambda_i(t_{I_m})\big) \tag{7.2}$$

and the estimation error is upper bounded by

$$|\Lambda_i(t) - \Lambda_i^{(m)}(t)| \leq {}^{O(\triangle t)e^{L_i t}}/_{L_i}$$

where $\lambda_i(t)$ is Lipschitz continuous with

$$|(\Lambda_i^{(m)})'(t) - \Lambda_i'(t)| \leq L_i|\Lambda_i^{(m)}(t) - \Lambda_i(t)|$$

and $O(\triangle t)$ denotes the first-order truncation error.

Using this numerical integration, it is straightforward to see that the log-likelihood in Eq. 7.1 can be approximately calculated regardless of the forms of intensity function $\mathcal{F}_i(\cdot)$. Next, we learn the parameters Θ by maximum likelihood estimation (MLE) using the gradient ascent method. Notably, we do not need to predefine the shape of the kernel functions. Thus the log-likelihood in Eq. 7.1 is concave, such that the global maximum and the convergence of inference can be guaranteed [52]. Additionally, Θ can be estimated in parallel over all dimensions.

LEMMA 7.2. *The per iteration computation cost of the parametric procedure is $O(M + \max\{\mathcal{I}_m\} \times m)$.*

7.2 Nonparametric Inference

Once the parameters Θ are estimated, we are left to estimate the kernel functions. The time shift in the kernel function $\phi_{ij}(t - t_{jl})$ in time domain corresponds to a multiplication by an exponential function in frequency domain as follows:

$$\phi_{ij}(t - t_{jl}) \Longrightarrow e^{-\mathrm{J}\omega t_{jl}}\mathbf{\Phi}_{ij}(\omega) \tag{7.3}$$

[‖] It is conventional to maximize the log of the likelihood function in order to handle underflow problem.

[**] We compare it with two other popular integration methods in terms of accuracy and convergence speed (see details in the Appendix). All these methods exhibit similar accuracy. As our modified Euler integration method shows the best efficiency under abundant activities, we shall adopt it by default in the performance study (Section 8).

It provides a way to simplify the time-shifted kernel functions, through which the intensity function $\lambda_i(t)$ can be transformed to the frequency domain, referred to as $\mathbf{\Lambda_i}(\omega)$.

If $\lambda_i(t)$ is a linear combination of a series of time-shifted kernel functions, it is straightforward to obtain $\mathbf{\Lambda_i}(\omega)$ as:

$$\begin{aligned}\mathbf{\Lambda_i}(\omega) &= \int_{-\infty}^{\infty} \lambda_i(t) e^{-\mathrm{J}\omega t} dt \\ &= \int_{-\infty}^{\infty} \left(\mu_i + \sum_{j\in[M]} \sum_{t_{jl}<t} \alpha_{ij}(t)\phi_{ij}(t-t_{jl})\right) e^{-\mathrm{J}\omega t} dt \\ &= 2\pi\mu_i\delta(\omega) + \sum_{j\in[M]} \alpha_{ij}(t) \sum_{t_{jl}<t} e^{-\mathrm{J}\omega t_{jl}} \mathbf{\Phi}_{ij}(\omega) \end{aligned} \tag{7.4}$$

Note that the nonlinear functions $\mathcal{F}_i(\cdot)$ prevent us from applying such Fourier transform directly. To circumvent this issue, we apply Taylor approximation to relax the linearity limitation, and derive the frequency domain counterpart of $\lambda_i(t)$ as following:

$$\begin{aligned}\mathbf{\Lambda_i}(\omega) &= \int_{-\infty}^{\infty} \lambda_i(t) e^{-\mathrm{J}\omega t} dt \\ &\approx \int_{-\infty}^{\infty} \left(\mathcal{F}_i(\mu_i) + \mathcal{F}_i'(\mu_i) \sum_{j\in[M]} \sum_{t_{jl}<t} \alpha_{ij}\phi_{ij}(t-t_{jl})\right) e^{-\mathrm{J}\omega t} dt \\ &= 2\pi\mathcal{F}_i(\mu_i)\delta(\omega) + \mathcal{F}_i'(\mu_i) \sum_{j\in[M]} \alpha_{ij}(t) \sum_{t_{jl}<t} e^{-\mathrm{J}\omega t_{jl}} \mathbf{\Phi}_{ij}(\omega) \end{aligned} \tag{7.5}$$

where $\delta(\omega)$ is the Dirac delta function[††].

According to Eq. 3.1, we could obtain that the expectation of an increment of the counting process $N_i(t+dt) - N_i(t)$ is essentially equivalent to $\lambda_i(t)dt$. Consequently, if we separate the period $(0, t]$ into N equal-length time slots (*i.e.,* $NT = t$) and denote the corresponding number of activities within each slot as $N_i[0], N_i[1], \ldots, N_i[N-1]$, respectively, $\mathbf{\Lambda_i}(\omega)$ can then be interpreted in terms of $N_i[k]$ as:

$$\mathbf{\Lambda_i}(\omega) = \sum_{k=0}^{N-1} N_i[k] e^{-\mathrm{J}\omega kT}$$

Since ω is a continuous variable, there are an infinite number of possible values of ω from 0 to 2π. Hence, $\mathbf{\Lambda_i}(\omega)$ could only be calculated at a finite set of frequencies. Therefore, we divide the unit circle into N equally area (*i.e.,* $\frac{1}{NT}$Hz, $\frac{2\pi}{NT}$rad/sec), and denote them as $\omega_n = \frac{2\pi}{NT} \times n$, then:

$$\mathbf{\Lambda_i}[n] = \sum_{k=0}^{N-1} N_i[k] e^{-\mathrm{J}\omega_n k} \qquad (n = 0 : N-1) \tag{7.6}$$

wherein $\mathbf{\Lambda_i}[n]$ contains information about the amplitude and phase of the sinusoid wave of frequency ω_n. Intuitively, the triggering kernel function $\phi_{ij}(t)$ should be proportional to the decay rate of previous interactions β_{ij} (estimated in Section 7.1). As a result, given ω_n, we could obtain :

$$\mathbf{\Phi}_{ij}[\omega_n] = \frac{\beta_{ij}\mathbf{\Lambda_i}[n]}{\mathcal{F}_i'(\mu_i) \sum_{j\in[M]} \alpha_{ij}(t)\beta_{ij} \sum_{l=1}^{N_j(t)} e^{-\mathrm{J}\omega_n t_{jl}}} \tag{7.7}$$

[††] $\delta(\omega)$ is zero everywhere except at $\omega = 0$, and its total integral is 1.

In particular,

$$\mathbf{\Phi}_{ij}[\omega_0] = \frac{\mathbf{\Lambda_i}[0] - 2\pi\mathcal{F}_i(\mu_i)}{\mathcal{F}_i'(\mu_i) \sum_{j\in[M]} \alpha_{ij}(t) N_j(t)} \tag{7.8}$$

Then we could deduce the time domain counterpart of $\mathbf{\Phi}_{ij}[\omega]$ by inversing DFT (IDFT):

$$\phi_{ij}(t) = \frac{1}{N} \sum_{n=0}^{N-1} \mathbf{\Phi}_{ij}[\omega_n] e^{\mathrm{J}\omega_n t} \tag{7.9}$$

The above estimation for kernel functions is completely data-driven, and $\int_0^{+\infty} t|\phi_{ij}(t)|dt < +\infty$. Then we could guarantee that Eq. 4.2 is stable in variation with respect to the initial condition $\lim_{s\to+\infty} \int_s^{+\infty} dt \int_{\mathcal{R}^-} |\phi_{ij}(t-u)|N_j(du) = 0$ according to Theorem 8 of [11].

We run the parametric and nonparametric inference procedures alternatively until convergence. For each iteration, DFT requires $O(N\log_2(N))$ operations. Since $\mathbf{\Phi}_{ij}[\omega_n]$ varies along the number of activities in $N_j(t)$, updating the kernel functions costs $O(\max\{N_j\} \times M)$ operations.

The semi-parametric inference procedure of CHASSIS is outlined in Algorithm 4. As mentioned earlier, it is identical to the M-step. It first initializes the parameters (Line 1), and then utilizes the diffusion trees in previous E-step to compute the two types of conformity-aware influence (Lines 2-4). Subsequently, it repeats the parametric estimation (Lines 6-13) and nonparametric estimation (Lines 14-20) till the loglikelihood reaches the predefined convergence tolerance.

7.3 Prediction of User Behaviors

By leveraging the estimated CHASSIS in the previous subsections, we design a procedure to predict the user behaviours.

Next activity prediction. We denote the timestamp of the most recent activity up to time t (including t) as t_n ($t_n \le t$), and the first activity after time t as t_{n+1}. Given the collection of activities $\mathbf{X}_t$ (*i.e.,* the historical activities $\mathcal{H}_{t_{n+1}} = \mathbf{X}_t$ until time t), the conditional density function of the occurrence time of the next activity t_{n+1} is:

$$\begin{aligned}\mathcal{P}_{n+1}(t) &= \mathcal{P}(t_{n+1} = t|\mathcal{H}_{t_{n+1}}) \\ &= \sum_{i\in[M]} \lambda_i(t) \prod_{i\in[M]} \exp\left(-\int_{t_n}^{t} \lambda_i(s)ds\right)\end{aligned} \tag{7.10}$$

Hence, we could predict the timestamp of the next activity according to the following expectation:

$$t_{n+1} = E\left[t_{n+1}|\mathcal{H}_{t_{n+1}}\right] = \int_{t_n}^{\infty} t\mathcal{P}_{n+1}(t)dt \tag{7.11}$$

In the following, we can predict which individual most likely generates that activity t_{n+1} via $\arg\max_{i\in[M]} \frac{\lambda_i(t)}{\sum_{i\in[M]}\lambda_i(t)}$. Suppose that U_i issues the activity t_{n+1}. We update the intensity $\{\lambda_i(t_{n+1})\}_{i\in[M]}$ with the informational influence and normative influence at time t (since we cannot predict the content of the activity t_{n+1}, we use the values $\alpha_{ij}^I(t), \alpha_{ij}^N(t)$ for all

Algorithm 4: Semi-parametric Estimation for CHASSIS.

Input : $\mathbf{X}_t = \{\{a_{ik} = (t_{ik}, C_{ik})\}_{k=1}^{N_i(t)}\}_{i=1}^{M}$
Output: $\{\mu_i, \gamma_{ij}^I(t), \beta_{ij}, \gamma_{ij}^N(t), \phi_{ij}(t)\}_{i,j\in[M]}$

Initialize $\{\mu_i, \gamma_{ij}^I(t), \beta_{ij}, \gamma_{ij}^N(t)\}_{i,j\in[M]}$;
$\{D_t\} \leftarrow$ DiffusionTreeConstruct($\mathbf{X}_t$);
/* Update the diffusion tress in the previous E-step */
$\boldsymbol{\alpha}^{\boldsymbol{I}}(\boldsymbol{t}) \leftarrow$ InformationalConform($\mathbf{X}_t$);
$\boldsymbol{\alpha}^{\boldsymbol{N}}(\boldsymbol{t}) \leftarrow$ NormativeConform($\mathbf{X}_t$);
while $\{\ln \mathcal{L}_i(\Theta|\mathbf{X}_t)\}_{i\in[M]}$ *not converged* **do**
  for $i \in \{1, \ldots, M\}$ **do**
    $\mu_i^{(r+1)} \leftarrow \max\left\{\mu_i^{(r)} + \eta_{r+1}\nabla_{\mu_i} \ln \mathcal{L}_i(\Theta^{(r)}|\mathbf{X}_t), 0\right\}$;
    for $j \in \{1, \ldots, M\}$ **do**
      $\beta_{ij}^{(r+1)} \leftarrow \beta_{ij}^{(r)} + \eta_{r+1}\nabla_{\beta_{ij}} \ln \mathcal{L}_i(\Theta^{(r)}|\mathbf{X}_t)$;
      $\left(\gamma_{ij}^I(t)\right)^{(r+1)} \leftarrow \left(\gamma_{ij}^I(t)\right)^{(r)} + \eta_{r+1}\nabla_{\gamma_{ij}^I(t)} \ln \mathcal{L}_i(\Theta^{(r)}|\mathbf{X}_t)$;
      $\left(\gamma_{ij}^N(t)\right)^{(r+1)} \leftarrow \left(\gamma_{ij}^N(t)\right)^{(r)} + \eta_{r+1}\nabla_{\gamma_{ij}^N(t)} \ln \mathcal{L}_i(\Theta^{(r)}|\mathbf{X}_t)$;
    end
  end
  for $i \in \{1, \ldots, M\}$ **do**
    $\{(\omega_n, \Lambda_i[n])\}_{n\in\{0,\ldots,N-1\}} \leftarrow \mathrm{DFT}(\lambda_i(t))$;
    for $j \in \{1, \ldots, M\}$ **do**
      Calculate $\{\mathbf{\Phi}_{ij}[\omega_n]\}_{n\in\{0,\ldots,N-1\}}$;
      $\phi_{ij}(t) \leftarrow \mathrm{IDFT}(\{\mathbf{\Phi}_{ij}[\omega_n]\}_{n\in\{0,\ldots,N-1\}})$;
    end
  end
end
return $\{\mu_i, \gamma_{ij}^I(t), \beta_{ij}, \gamma_{ij}^N(t), \phi_{ij}(t)\}_{i,j\in[M]}$

$i, j \in [M]$) accordingly. Then, we can check whether the activity t_{n+1} is an immigrant or an offspring via reconstructing the diffusion trees of the new collection of activities (*i.e.,* $\mathbf{X}_t$ plus t_{n+1}).

Future number of activities prediction. Predicting the future number of activities has broad applications, such as identifying potentially viral messages before they become popular, and forecasting the effect of external interventions. Given the observations $\mathbf{X}_t$ up to time t, prediction of the number of activities from t to a future time point $\bar{t} > t$ can be carried out as follows. We simulate the processes $\{N_i\}_{i\in[M]}$ associated with the intensity functions in Eq. 4.2 over the interval $(t, \bar{t}]$ conditional on $\mathbf{X}_t$, and then calculate the average of the simulated activities. Intuitively, it is computationally expensive to generate a larger number of activities for such

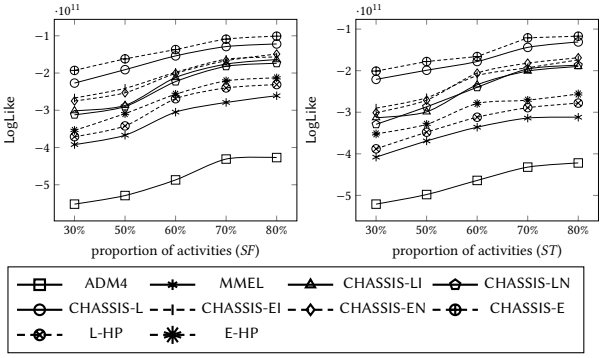


Figure 5. Model fitness (LogLike).

simulation. Hence, we estimate the expected number of activities with the expectation of the intensity conditional on $\mathbf{X}_t$ as follows:

$$\mathbb{E}\Big[\sum_{i\in[M]}(N_i(\bar{t}) - N_i(t)|\mathcal{H}_t\Big] = \mathbb{E}\Big[\sum_{i\in[M]}\int_0^{\bar{t}-t}\lambda_i(s)ds|\mathcal{H}_t\Big]$$
$$= \sum_{i\in[M]}\int_0^{\bar{t}-t}\mathbb{E}[\lambda_i(s)|\mathcal{H}_t]ds = \sum_{i\in[M]}\int_0^{\bar{t}-t}\overline{\lambda_i}(s)ds \quad (7.12)$$

Theorem 7.3. *Suppose $\overline{\lambda}_i(s)$ is the mean intensity conditional on $\mathbf{X}_t$ over $[0, \bar{t}-t]$. Then under the Taylor approximation $\lambda_i(t) \approx \mathcal{F}_i(\mu_i) + \mathcal{F}_i'(\mu_i)\sum_{j\in[M]}\sum_{t_{jl}<t}\alpha_{ij}(t)\phi_{ij}(t-t_{jl})$, $\overline{\lambda}_i(s)$ satisfies the following* Volterra integral equation*:*

$$\overline{\lambda}_i(s) = \mathcal{F}_i(\mu_i) + \mathcal{F}_i'(\mu_i)\sum_{j\in[M]}\alpha_{ij}(t)\int_0^s \phi_{ij}(s-\tau)\overline{\lambda}_j(\tau)d\tau$$

We solve the Volterra integral equation in Theorem 7.3 numerically via the trapezoidal rule [48]. Then we can obtain the conditional expectation of the future number of activities according to Eq. 7.12, which is more efficient than the costly simulation of a large number of activities.

8 PERFORMANCE STUDY

In the section, we demonstrate the performance of CHASSIS. We have implemented the framework in Python. All experiments are performed on a 64-bit Windows desktop with Intel(R) Core(TM) E5-1620V2 CPU@3.70 and 16GB RAM.

Strategies. We compare the following information diffusion models that are most germane to our work:

- NetRate [23]: A popular information diffusion modeling technique in social networks. Note that we only utilize the first activity of each individual because it cannot model the recurrent activities.
- ADM4 [60]: It utilizes the mutually-exciting linear Hawkes model to capture temporal patterns of user behaviors, and infers the social influence by imposing both low-rank and sparse regularization on the influence matrix.

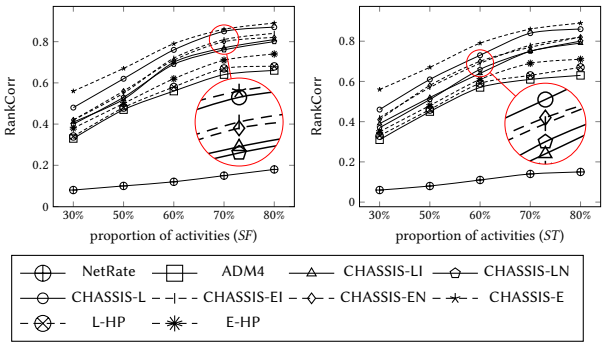


Figure 6. Model fitness (RankCorr).

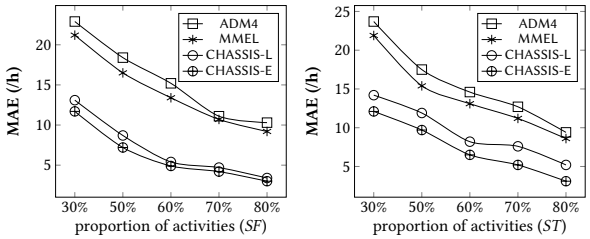


Figure 7. Future number of activities prediction.

- MMEL [61]: It captures the temporal dynamics of observed activities by linear Hawkes processes, and learns the triggering kernels nonparametrically.
- CHASSIS-L: Our proposed CHASSIS model. Here, we set $\mathcal{F}_i(x) = x$ (*i.e.,* linear Hawkes processes). Initially, the base intensity μ is sampled from a uniform distribution over $[0, 0.01]$ for each dimension, and the coefficients $\{\gamma_{ij}^I(t), \beta_{ij}, \gamma_{ij}^N(t)\}_{i,j\in[M]}$ are generated from a uniform distribution on $[0, 0.1]$.
- CHASSIS-E: Similar to CHASSIS-L, the only difference is $\mathcal{F}_i(x) = e^x$ (*i.e.,* exponential Hawkes processes).

Remark. (a) NetRate is an information diffusion model in continuous time domain (*i.e.,* not based on Hawkes processes); (b) Two Hawkes-based information diffusion models, one with parametric inference method (ADM4) and the other with semi-parametric inference method (MMEL). Both the inference procedures involve the branching structure.

8.1 Model Fitness and Prediction

Datasets. We use the following datasets: (a) *Facebook*: We collect the data via Facebook Graph API (https://developers.facebook.com/docs/graph-api), comprising nearly 44 million public activities posted by 109, 211 individuals, from March 2018 to May 2018; (b) *Twitter*: We gather the data via Twitter Streaming API (https://developer.twitter.com/en/docs), containing nearly 52 million public activities posted by 123, 972 individuals, from March 2018 to May 2018. Additionally, we obtain the relationships among such individuals (*i.e.,* who follows whom) in each dataset, which could be converted into an excitation matrix $\boldsymbol{A} = [\alpha_{ij}]_{i,j\in[M]}$ ($\alpha_{ij} = 1$ if U_j follows U_i, otherwise $\alpha_{ij} = 0$) as the ground truth. Utilizing such relationships, we grab the offspring activities of each immigrant activity of each individual via a depth first search algorithm. We evaluate the scalability of CHASSIS using these two datasets and extract two subsets of the datasets: 590,671 activities posted by 100,000 individuals in *Facebook* (denoted as *SF*) and the other with 671,810 activities posted by 110,000 individuals in *Twitter* (denoted as *ST*), for other experiments.

Model Fitness. We study how well CHASSIS can explain the real-world data by comparing it with other strategies. We use two evaluation metrics, namely LogLike and RankCorr. Specifically, LogLike is the log-likelihood of the estimated model on one test dataset and computed as $\ln \mathcal{L}(\mathbf{X}_{\text{test}}|\Theta_{\text{training}}) = \sum_{i\in[M]} \left(\sum_{k=1}^{N_i(t_{\text{test}})} \ln \lambda_i(t_{ik}) - \int_0^{t_{\text{test}}} \lambda_i(s)ds \right)$ [60]. RankCorr calculates the average Kendall's rank correlation coefficient between each row of *influence matrix* $\boldsymbol{A}$ and estimated $\hat{\boldsymbol{A}}$, to measure whether the relative order of the estimated social influences is correctly recovered [60]. We order all activities in a dataset chronologically, and use the first 30%, 50%, 60%, 70%, 80% samples for training, respectively.

Figure 5 shows the performance using LogLike on the testing activities. Note that we exclude NetRate as it could not model the recurrent activities in our data. Observe that LogLike increases as the number of activities for training increases, indicating that more training data lead to better accuracy for all approaches. Clearly, CHASSIS-L and CHASSIS-E perform significantly better than ADM4 and MMEL, which indicates that CHASSIS can capture the information diffusion better than the conformity-unaware strategies.

We are also interested in clarifying whether the superiority of CHASSIS is due to conformity-awareness or merely more flexible semi-parametric inference method. To this end, we design two baselines, L-HP and E-HP, referring to our semi-parametric inference algorithm under linear and exponential Hawkes, respectively, with the intensities in Eq. 3.2. Notably, both methods are conformity-unaware. As shown in Figure 5, both are inferior to CHASSIS-L and CHASSIS-E. Hence, model fitness accuracy can be improved significantly when conformity is taken into account. On the other hand, both baselines exhibit better performance than ADM4 and MMEL. It implies that the proposed semi-parametric inference scheme also improves model fitness performance.

Additionally, we investigate the importance of modeling both informational and normative conformity in CHASSIS by disabling one of them in Eq. 4.2. Specifically, we remove $\sum_{i,j\in[M]} \gamma_{ij}^N(t)\alpha_{ij}^N(t)$ (resp. $\sum_{i,j\in[M]} \gamma_{ij}^I(t)\alpha_{ij}^I(t)$), and only quantify $\sum_{i,j\in[M]} \gamma_{ij}^I(t)\alpha_{ij}^I(t)$ (resp. $\sum_{i,j\in[M]} \gamma_{ij}^N(t)\alpha_{ij}^N(t)$) in CHASSIS-LI (resp. CHASSIS-LN) and CHASSIS-EI (resp.

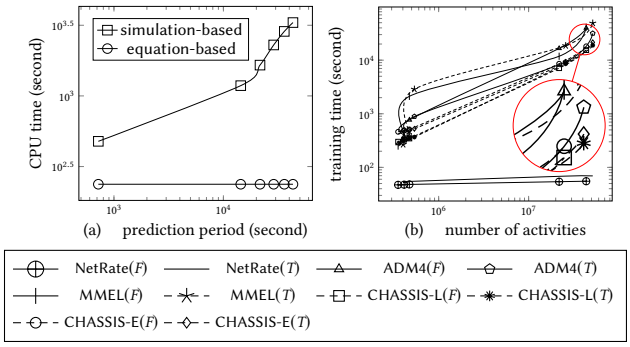


Figure 8. Prediction and scalability (F=*Facebook*, T=*Twitter*).

CHASSIS-EN). As shown in Figure 5, CHASSIS-L (resp. CHASSIS-E) outperforms CHASSIS-LI (resp. CHASSIS-EI) and CHASSIS-LN (resp. CHASSIS-EN), endorsing the necessity of modeling both informational conformity and normative conformity. All these approaches also outperform L-HP and E-HP. Furthermore, CHASSIS-E (resp. CHASSIS-EI, CHASSIS-EN and E-HP) always has higher LogLike than CHASSIS-L (resp. CHASSIS-LI, CHASSIS-LN and L-HP), indicating that nonlinear Hawkes processes are more suitable for capturing the triggering patterns hidden in social activities.

The results using RankCorr (Figure 6) are qualitatively similar to LogLike. The nonparametric estimation in MMEL embeds the influence matrix into the triggering kernels, thus, RankCorr cannot be calculated for it.

In summary, CHASSIS fits the real-world data better compared to conformity-unaware information diffusion models.

Prediction of Activities. We compare the performance in predicting the future number of activities w.r.t the window size $\triangle_t = \bar{t} - t$. Note that since NetRate fail to model recurrent activities, it cannot predict the future number of activities. The mean intensity $\overline{\lambda_i}(s)$ is evaluated by solving the equation in Theorem 7.3 for all Hawkes-based strategies. Accordingly, the number of future activities during each future time window is calculated from $N_i^k = \sum_{i \in [M]} \int_{t+(k-1)\triangle_t}^{t+k\triangle_t} \overline{\lambda_i}(s) ds$. The *mean absolute error* (MAE) is defined as

$$\zeta = \frac{1}{M} \sum_{i \in [M]} \frac{1}{t_{\max} - t} \sum_{k \in [\frac{t_{\max}-t}{\triangle_t}]} |\hat{N}_i^k - N_i^k|$$

where M is the total number of individuals, N_i^k and $\hat{N}_i^k$ are the actual and predicted number of activities in the k^{th} time window, and $t_{\max}$ is the end time of the prediction period.

Figure 7 shows that error decreases with the increase in training data. In particular, it decreases quickly for ADM4 and MMEL. This confirms the superiority of Hawkes processes in modeling information diffusion. More importantly, both CHASSIS-E and CHASSIS-L provide the best prediction performance for all cases, indicating that conformity awareness significantly improves the prediction accuracy.

Figure 8(a) shows in log-log scale the time cost *w.r.t* the prediction period. Comparatively, our equation-based approach in Theorem 7.3 is more efficient than the simulation-based approach (please refer to [45] for simulation details).

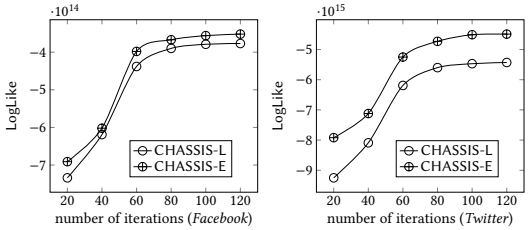


Figure 9. Convergence study (LogLike).

Convergence. We demonstrate the convergence rate of CHASSIS on *Facebook* and *Twitter* datasets. Figure 9 reports the results using LogLike. Clearly, the LogLike of CHASSIS-L and CHASSIS-E increase as the number of iterations grows and converge after 80 iterations on *Facebook* data. The performance on *Twitter* data is qualitatively similar.

Scalability. We compare the scalability of CHASSIS with other models using the training time (*i.e.,* model inference time) on *Facebook* and *Twitter* datasets. Figure 8(b) plots the training time in log-log scale. Clearly, the computation cost of CHASSIS-L and CHASSIS-E scales almost linearly with the number of activities. Except for NetRate, CHASSIS performs the best. Since NetRate only deals with the first activity of each individual, its training time is naturally faster.

8.2 Diffusion Tree Inference

We now report the diffusion tree inference quality in CHASSIS using the datasets from PHEME (https://doi.org/10.6084/m9.figshare.6392078.v1) [32]. It contains *Twitter* conversation threads (*i.e.,* information cascades) associated with different newsworthy topics including the Ferguson unrest, the shooting at Charlie Hebdo, the shooting in Ottawa, the hostage situation in Sydney and the crash of a Germanwings plane. Table 2 shows the statistics. Each conversation consists of a source tweet conveying a rumour or a non-rumour and a tree of responses, expressing their opinions toward the claim contained in the source tweet. Since the diffusion trees in each conversation are given, we use them as the ground truth for evaluating the inferred diffusion trees.

Table 2. Statistics of the PHEME dataset.

Topic	#Cascades	#Activities
Charlie Hebdo	2,079	38,268
Sydney siege	1,221	23,996
Ferguson	1,143	24,175
Ottawa shooting	890	12,284
Germanwings-crash	469	4,489

Table 3. Branching structure inference performance

Dataset	Strategy			
	ADM4	MMEL	CHASSIS-L	CHASSIS-E
Charlie Hebdo	0.6547	0.7031	0.7966	0.8422
Sydney siege	0.6301	0.6908	0.7804	0.8380
Ferguson	0.6122	0.6743	0.7765	0.8201
Ottawa shooting	0.6003	0.6578	0.7523	0.8130
Germanwings-crash	0.5634	0.6020	0.7342	0.8002

Inferring the Diffusion Trees. ADM4 and MMEL could infer the branching structure by utilizing the linear accumulation of triggering kernels [60, 61]. We run the four Hawkes-based strategies on each dataset, and construct the diffusion trees accordingly. We store the diffusion trees in a binary matrix wherein the index of the rows and the columns are the chronologically ordered activities. Comparing the inferred branching structure with the ground truth, we evaluate the effectiveness of all the aforementioned strategies in terms of F1-Score. Table 3 reports the results. Observe that more activities (*i.e.,* larger dataset) naturally improve the inference accuracy. More importantly, both CHASSIS-E and CHASSIS-L outperform the conformity-unaware strategies on all datasets, reemphasizing the importance of conformity in information diffusion. Again, CHASSIS-E outperforms CHASSIS-L significantly, indicating that nonlinear Hawkes processes are more appropriate than the linear ones.

Next Activity Prediction. We use the first 90% of each dataset for training and keep the last 10% for testing. Given the estimated Hawkes-based models (excluding NetRate), we predict the timestamp of the next activity, as well as the individual who will issue it according to Section 7.3. Notably, whenever we have predicted a next activity a, it is evaluated against the ground truth one. Afterwards, we append this ground truth activity to the training set in order to update the corresponding intensity functions (including the ground-truth parent-child relationship if it is an immigrant in CHASSIS), and then use the updated intensity functions to predict another new activity. We perform this process iteratively over all tested samples. Finally, we compute the MAE of the timestamp of the next activity according to $\epsilon = \frac{1}{N(t_{\text{test}})} \sum_{i \in [N(t_{\text{test}})]} |\hat{t}_i - t_i|$, where $N(t_{\text{test}})$ is the number of tested samples in each dataset, $\hat{t}_i$ and t_i are the predicted and actual timestamp of the next activity. Additionally, we calculate the average precision of predicted individual who will generate the next activity according to $\frac{\sum_{k \in [N(t_{\text{test}})]} \mathbb{I}(a_k \in \mathbf{X}_i(t_{\text{test}}) \wedge \hat{a}_k \in \mathbf{X}_i(t_{\text{test}}))}{N(t_{\text{test}})}$, where the indicator function $\mathbb{I}(\cdot)$ is to decide whether the prediction of the individual generating the predicted activity $\hat{a}_k$ is correct or not.

Figure 10(a) plots the MAE for predicting the timestamp of the next activity. Clearly, CHASSIS-L and CHASSIS-E outperform the baselines. Figure 10(b) shows the precision for predicting which individual will generate the next activity.

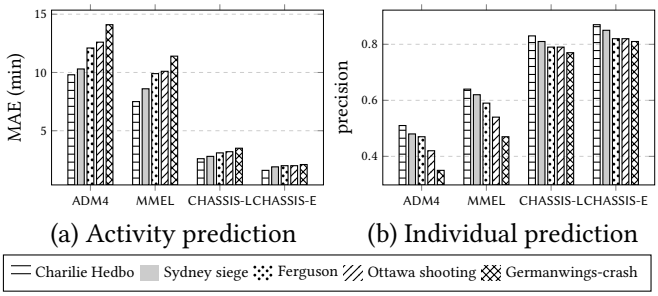


Figure 10. Prediction performance of the next activity.

CHASSIS-E is again the best predictor. The intensity update of ADM4 does not involve the information of branching structure, hence, it has the worst performance in both prediction scenarios. MMEL utilizes the branching structure while updating the triggering kernels nonparametrically.

9 CONCLUSIONS

A significant omission in existing online information diffusion models is the role played by conformity, a fundamental human trait according to social psychology theories. We propose a novel framework called CHASSIS to address this limitation by integrating informational conformity and normative conformity into Hawkes process-based information diffusion model. Specifically, we detect and quantify conformity by analyzing the triggering relations among the activities represented as diffusion trees. We propose an efficient semi-parametric inference algorithm, wherein the parametric evaluation procedure assists in identifying conformity of individuals, and the nonparametric procedure learns the triggering kernel functions flexibly in a data-driven way without the need of prior domain knowledge. Our experimental study not only demonstrates superiority of CHASSIS compared to conformity-unaware models but also emphasizes the pivotal role conformity plays in information diffusion.

REFERENCES

[1] Rediet Abebe, Jon M. Kleinberg, David C. Parkes, and Charalampos E. Tsourakakis. 2018. Opinion Dynamics with Varying Susceptibility to Persuasion. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018.* 1089–1098.

[2] Lisa R Anderson and Charles A Holt. 2006. Information cascades and rational conformity. *Encyclopedia of Cognitive Science* (2006).

[3] E. Aronson, T.D. Wilson, and R.M. Akert. 2007. *Social Psychology.* Pearson Education International.

[4] Akhil Arora, Sainyam Galhotra, and Sayan Ranu. 2017. Debunking the Myths of Influence Maximization: An In-Depth Benchmarking Study. In *Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, Chicago, IL, USA, May 14-19, 2017.* 651–666.

[5] Solomon E Asch. 1951. Effects of group pressure upon the modification and distortion of judgments. *Organizational influence processes* (1951), 295–303.

[6] Solomon E Asch. 1955. Opinions and social pressure. *Scientific American* 193, 5 (1955), 31–35.

[7] Prithu Banerjee, Wei Chen, and Laks V. S. Lakshmanan. 2019. Maximizing Welfare in Social Networks under A Utility Driven Influence Diffusion model. In *Proceedings of the 2019 International Conference on Management of Data, SIGMOD Conference 2019, Amsterdam, The Netherlands, June 30 - July 5, 2019*. 1078–1095.

[8] Kshipra Bhawalkar, Sreenivas Gollapudi, and Kamesh Munagala. 2013. Coevolutionary opinion formation games. In *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*. 41–50.

[9] Sushil Bikhchandani, David Hirshleifer, and Ivo Welch. 1992. A theory of fads, fashion, custom, and cultural change as informational cascades. *Journal of political Economy* 100, 5 (1992), 992–1026.

[10] Steven J Breckler, James Olson, and Elizabeth Wiggins. 2005. *Social psychology alive*. Cengage Learning.

[11] Pierre Brémaud and Laurent Massoulié. 1996. Stability of nonlinear Hawkes processes. *The Annals of Probability* (1996), 1563–1588.

[12] Jennifer D Campbell and Patricia J Fairey. 1989. Informational and normative routes to conformity: The effect of faction size as a function of norm extremity and attention to the stimulus. *Journal of personality and social psychology* 57, 3 (1989), 457.

[13] Shuo Chen, Ju Fan, Guoliang Li, Jianhua Feng, Kian-Lee Tan, and Jinhui Tang. 2015. Online Topic-Aware Influence Maximization. *PVLDB* 8, 6 (2015), 666–677.

[14] Justin Cheng, Lada A. Adamic, P. Alex Dow, Jon M. Kleinberg, and Jure Leskovec. 2014. Can cascades be predicted?. In *23rd International World Wide Web Conference, WWW '14, Seoul, Republic of Korea, April 7-11, 2014*. 925–936.

[15] Nicolas Claidière and Andrew Whiten. 2012. Integrating the study of conformity and culture in humans and nonhuman animals. *Psychological bulletin* 138, 1 (2012), 126.

[16] William D Crano. 1970. Effects of sex, response order, and expertise in conformity: A dispositional approach. *Sociometry* (1970), 239–252.

[17] Abhimanyu Das, Sreenivas Gollapudi, Arindam Khan, and Renato Paes Leme. 2014. Role of conformity in opinion dynamics in social networks. In *Proceedings of the second ACM conference on Online social networks, COSN 2014, Dublin, Ireland, October 1-2, 2014*. 25–36.

[18] Philip J Davis and Philip Rabinowitz. 2007. *Methods of numerical integration*. Courier Corporation.

[19] Morton Deutsch and Harold B Gerard. 1955. A study of normative and informational social influences upon individual judgment. *The journal of abnormal and social psychology* 51, 3 (1955), 629.

[20] Kuntal Dey, Ritvik Shrivastava, and Saroj Kaushik. 2017. Twitter Stance Detection - A Subjectivity and Sentiment Polarity Inspired Two-Phase Approach. In *2017 IEEE International Conference on Data Mining Workshops, ICDM Workshops 2017, New Orleans, LA, USA, November 18-21, 2017*. 365–372.

[21] Sainyam Galhotra, Akhil Arora, and Shourya Roy. 2016. Holistic Influence Maximization: Combining Scalability and Efficiency with Opinion-Aware Models. In *Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016, San Francisco, CA, USA, June 26 - July 01, 2016*. 743–758.

[22] Jacob K Goeree, Thomas R Palfrey, Brian W Rogers, and Richard D McKelvey. 2007. Self-correcting information cascades. *The Review of Economic Studies* 74, 3 (2007), 733–762.

[23] Manuel Gomez-Rodriguez, David Balduzzi, and Bernhard Schölkopf. 2011. Uncovering the Temporal Dynamics of Diffusion Networks. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*. 561–568.

[24] Manuel Gomez-Rodriguez, Jure Leskovec, and Andreas Krause. 2010. Inferring networks of diffusion and influence. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, July 25-28, 2010*. 1019–1028.

[25] Caitlin Gray, Lewis Mitchell, and Matthew Roughan. 2018. Superblockers and the Effect of Network Structure on Information Cascades. In *Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*. 1435–1441.

[26] Adrien Guille, Hakim Hacid, Cécile Favre, and Djamel A. Zighed. 2013. Information diffusion in online social networks: a survey. *SIGMOD Record* 42, 2 (2013), 17–28.

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

[28] Alan G Hawkes and David Oakes. 1974. A cluster process representation of a self-exciting process. *Journal of Applied Probability* 11, 3 (1974), 493–503.

[29] Herbert W. Hethcote. 2000. The Mathematics of Infectious Diseases. *SIAM Rev.* 42, 4 (2000), 599–653.

[30] Harold H Kelley and Martin M Shapiro. 1954. An experiment on conformity to group norms where conformity is detrimental to group achievement. *American Sociological Review* 19, 6 (1954), 667–677.

[31] David Kempe, Jon M. Kleinberg, and Éva Tardos. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 24 - 27, 2003*. 137–146.

[32] Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. 2018. PHEME dataset for Rumour Detection and Veracity Classification.

[33] Dong Li, Shengping Zhang, Xin Sun, Huiyu Zhou, Sheng Li, and Xuelong Li. 2017. Modeling Information Diffusion over Social Networks for Temporal Dynamic Prediction. *IEEE Trans. Knowl. Data Eng.* 29, 9 (2017), 1985–1997.

[34] Guoliang Li, Shuo Chen, Jianhua Feng, Kian-Lee Tan, and Wen-Syan Li. 2014. Efficient location-aware influence maximization. In *International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22-27, 2014*. 87–98.

[35] Hui Li, Sourav S. Bhowmick, and Aixin Sun. 2011. CASINO: towards conformity-aware social influence analysis in online social networks. In *Proceedings of the 20th ACM Conference on Information and Knowledge Management, CIKM 2011, Glasgow, United Kingdom, October 24-28, 2011*. 1007–1012.

[36] Hui Li, Sourav S. Bhowmick, Aixin Sun, and Jiangtao Cui. 2015. Conformity-aware influence maximization in online social networks. *VLDB J.* 24, 1 (2015), 117–141.

[37] Yuchen Li, Ju Fan, Dongxiang Zhang, and Kian-Lee Tan. 2017. Discovering Your Selling Points: Personalized Social Influential Tags Exploration. In *Proceedings of the 2017 ACM International Conference on Management of Data, SIGMOD Conference 2017, Chicago, IL, USA, May 14-19, 2017*. 619–634.

[38] Yiqing Li, Xiaoying Gan, Luoyi Fu, Xiaohua Tian, Zhida Qin, and Yanhong Zhou. 2018. Conformity-Aware Influence Maximization with User Profiles. In *10th International Conference on Wireless Communications and Signal Processing, WCSP 2018, Hangzhou, China, October 18-20, 2018*. 1–6.

[39] Scott W. Linderman and Ryan P. Adams. 2014. Discovering Latent Network Structure in Point Process Data. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*. 1413–1421.

[40] Linyuan Lu, Duanbing Chen, and Tao Zhou. 2011. Small world yields the most effective information spreading. *CoRR* abs/1107.0429 (2011).

[41] Michal Lukasik, Trevor Cohn, and Kalina Bontcheva. 2015. Point Process Modelling of Rumour Dynamics in Social Media. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL*

2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers. 518–523.
[42] Hongyuan Mei and Jason Eisner. 2017. The Neural Hawkes Process: A Neurally Self-Modulating Multivariate Point Process. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (Eds.). 6754–6764.
[43] Mark E. J. Newman. 2003. The Structure and Function of Complex Networks. *SIAM Rev.* 45, 2 (2003), 167–256.
[44] Hung T. Nguyen, My T. Thai, and Thang N. Dinh. 2016. Stop-and-Stare: Optimal Sampling Algorithms for Viral Marketing in Billion-scale Networks. In *Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016, San Francisco, CA, USA, June 26 - July 01, 2016*. 695–710.
[45] Yosihiko Ogata. 1981. On Lewis' simulation method for point processes. *IEEE Trans. Information Theory* 27, 1 (1981), 23–30.
[46] Sen Pei, Lev Muchnik, Shaoting Tang, Zhiming Zheng, and Hernán A. Makse. 2015. Exploring the complex pattern of information spreading in online blog communities. *CoRR* abs/1504.00495 (2015).
[47] Chanthika Pornpitakpan. 2004. The persuasiveness of source credibility: A critical review of five decades' evidence. *Journal of applied social psychology* 34, 2 (2004), 243–281.
[48] William H Press, Saul A Teukolsky, William T Vetterling, and Brian P Flannery. 2007. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press.
[49] Liudmila Ostroumova Prokhorenkova, Alexey Tikhonov, and Nelly Litvak. 2019. Learning Clusters through Information Diffusion. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*. 3151–3157.
[50] Jie Tang, Sen Wu, and Jimeng Sun. 2013. Confluence: conformity influence in large social networks. In *The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013, Chicago, IL, USA, August 11-14, 2013*. 347–355.
[51] Youze Tang, Yanchen Shi, and Xiaokui Xiao. 2015. Influence Maximization in Near-Linear Time: A Martingale Approach. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 - June 4, 2015*. 1539–1554.
[52] William Trouleau, Jalal Etesami, Matthias Grossglauser, Negar Kiyavash, and Patrick Thiran. 2019. Learning Hawkes Processes Under Synchronization Noise. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. 6325–6334.
[53] Maria Nicolette Margaretha vanLieshout. 2006. Campbell and moment measures for finite sequential spatial processes. *CWI. Probability, Networks and Algorithms [PNA]* R 0601 (2006).
[54] David Vere-Jones. 2003. *An Introduction to the Theory of Point Processes: Volume I: Elementary Theory and Methods*. Springer.
[55] Ke Wu, Song Yang, and Kenny Q. Zhu. 2015. False rumors detection on Sina Weibo by propagation structures. In *31st IEEE International Conference on Data Engineering, ICDE 2015, Seoul, South Korea, April 13-17, 2015*. 651–662.
[56] Shuang-Hong Yang and Hongyuan Zha. 2013. Mixture of Mutually Exciting Processes for Viral Diffusion. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*. 1–9.
[57] Linyun Yu, Peng Cui, Fei Wang, Chaoming Song, and Shiqiang Yang. 2015. From Micro to Macro: Uncovering and Predicting Information Cascading Process with Behavioral Dynamics. In *2015 IEEE International Conference on Data Mining, ICDM 2015, Atlantic City, NJ, USA, November 14-17, 2015*. 559–568.
[58] Fue Zeng, Ran Tao, Yanwu Yang, and Tingting Xie. 2017. How social communications influence advertising perception and response in online communities? *Frontiers in psychology* 8 (2017), 1349.
[59] Jing Zhang, Jie Tang, Honglei Zhuang, Cane Wing-Ki Leung, and Juan-Zi Li. 2014. Role-Aware Conformity Modeling and Analysis in Social Networks. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada*. 958–965.
[60] Ke Zhou, Hongyuan Zha, and Le Song. 2013. Learning Social Infectivity in Sparse Low-rank Networks Using Multi-dimensional Hawkes Processes. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2013, Scottsdale, AZ, USA, April 29 - May 1, 2013*. 641–649.
[61] Ke Zhou, Hongyuan Zha, and Le Song. 2013. Learning Triggering Kernels for Multi-dimensional Hawkes Processes. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*. 1301–1309.

Appendices

9.1 Proofs

Proof of Lemma 5.1 (Sketch). The Line 5 and Line 9 of Algorithm 1 compute the influence degree and context stance for each parent-child pair of activities in $N_{ij}(t)$, respectively. Suppose the maximum number of parent-child activity pairs in $\{N_{ij}\}_{i,j\in[M]}$ is N_{max}. Hence, the computation cost of Algorithm 1 is $O(M^2 N_{max})$, where M is the number of dimensions (*i.e.,* individuals). ∎

Proof of Lemma 5.2 (Sketch). We denote the number of activities in one cascade c ($c \in C_{ij}$) as $|c|$. At most c has $|c|(|c|-1)$ pairs of activities that satisfy a_{jl} happens before a_{ik}. So the worse-case time complexity of Algorithm 2 is $O(M^2 m^2 n)$, where n the number of informational cascades in C_{ij} , $m = \max |C_{ij}| \ll M$ is the maximum number of activities in a single cascade and M is the number of dimensions (*i.e.,* individuals). ∎

Proof of Lemma 6.1 (Sketch). After each iteration of semi-parametric inference of CHASSIS, we need to update the intensity shown in Eq. 4.2 (Line 2 of Algorithm 3). Given one activity a_{ik}, we need to invert the corresponding Papangelous conditional intensity (Line 4 and 7) to update the parent-child activity pairs. Then the worst-case time complexity is to infer the parent activity of the last activity in X_t. Hence, Algorithm 3 leads to a $O(N_{\text{iter}} n^2)$ computation, where N_{iter} is the number of iterations required by the inference of CHASSIS and n is the total number of activities in X_t. ∎

Proof of Theorem 7.1 (Sketch). We discretize the observation window $(0, t]$ into I intervals of equal length $h = \frac{t}{I}$. Let $t_k = kh$ for $k = 0, 1, \ldots, I$. Then we define the following recursive sequence:

$$\Lambda_i(t_{k+1}) = \Lambda_i(t_k) + h\Lambda_i'(t_k) \tag{9.1}$$

wherein $\Lambda_i(t_k) = \int_0^{t_k} \lambda_i(s)ds$. According to Taylor's Theorem, each recursion generates a first-order truncation error as follows:

$$e = O(\tau) = \frac{h^2}{2}\Lambda_i''(\tau) = \frac{h^2}{2}\lambda_i'(\tau) \tag{9.2}$$

wherein $\frac{\lambda_i'(\tau)}{2}$ is assumed to be constant. As a consequence, given the interval length h, the accumulated truncation error after all recursions equals to:

$$E = \sum_{k=1}^{\mathcal{I}} e = \mathcal{I} \cdot \frac{h^2}{2}\lambda_i'(\tau) = \frac{t}{h} \cdot \frac{h^2}{2}\lambda_i'(\tau) = \frac{t\lambda_i'(\tau)}{2}h \tag{9.3}$$

Next, we iterate the above approximation calculation in Eq. 9.1 with more intervals (*i.e.,* lager I, smaller h):

$$\Lambda_i^{(m)}(t_{k+1}) = \Lambda_i^{(m)}(t_k) + h_m\lambda_i(t_k) \tag{9.4}$$

until reaching a prescribed error tolerance ξ. That is to say, we devise a sequence h_m ($m = 1, 2, \ldots$) until $|\Lambda_i^{(m+1)}(t_{I_{m+1}}) - \Lambda_i^{(m)}(t_{I_m})|$ becomes small enough. We deduce the interval length h_{m+1} of next iteration according to the accumulated truncation error $E_m = \frac{th_m}{2}\lambda_i'(\tau)$ and $E_{m-1} = \frac{th_{m-1}}{2}\lambda_i'(\tau)$ as following:

$$\Lambda_i^{(m)}(t_{I_m}) - \Lambda_i^{(m-1)}(t_{I_{m-1}}) = \frac{t\lambda_i'(\tau)}{2}(h_m - h_{m-1}) \tag{9.5}$$

Accordingly, the next iteration will produce an truncation error within ξ if

$$\xi \geq |\frac{t\lambda_i'(\tau)}{2}h_{m+1}| = |\frac{\Lambda_i^{(m)}(t_{I_m}) - \Lambda_i^{(m-1)}(t_{I_{m-1}})}{(h_m - h_{m-1})}h_{m+1}|$$

So, the next interval length h_{m+1} is determined by

$$h_{m+1} \leq \frac{\xi(h_{m-1} - h_m)}{|\Lambda_i^{(m)}(t_{I_m}) - \Lambda_i^{(m-1)}(t_{I_{m-1}})|}$$

Concretely, we yield the adaptive interval length iteratively

$$h_{m+1} = \frac{\rho\xi(h_{m-1} - h_m)}{|\Lambda_i^{(m)}(t_{I_m}) - \Lambda_i^{m-1}(t_{I_{m-1}})|} \tag{9.6}$$

for some coefficient $\rho < 1$ to reduce the total number of iterations. Obviously, $\Lambda_i^{(m+1)}(0) = h_{m+1}\mu_i$. According to the recursive sequence in Eq. 9.1, the $(m+1)^{\text{th}}$ iteration yields an approximation of $\int_0^t \lambda_i(s)ds$ as follows:

$$\Lambda_i^{(m+1)}(t) = h_{m+1}\Big(\mu_i + \lambda_i(t_1) + \ldots + \lambda_i(t_{I_{m+1}})\Big) \tag{9.7}$$

Below, we characterize the accuracy of the above Euler method. At time t_k in the m^{th} iteration, we obtain the real truncation error:

$$e^{(m)} = \frac{\Lambda_i^{(m)}(t_{k+1}) - \Lambda_i^{(m)}(t_k)}{h_m} - (\Lambda^{(m)})'(t_k) \tag{9.8}$$

Rearrange the above equation:

$$\Lambda_i^{(m)}(t_{k+1}) = \Lambda_i^{(m)}(t_k) + h_m(\Lambda^{(m)})'(t_k) + h_m e^{(m)} \tag{9.9}$$

Refer to our definition in Eq.9.1:

$$\Lambda_i(t_{k+1}) = \Lambda_i(t_k) + h_m\Lambda_i'(t_k) \tag{9.10}$$

$\lambda_i(t)$ is Lipschitz continuous, hence,

$$|\frac{(\Lambda_i^{(m)})'(t) - \Lambda_i'(t)}{\Lambda_i^{(m)}(t) - \Lambda_i(t)}| \leq L_i \tag{9.11}$$

Subtract Eq.9.9 from Eq.9.10, and then substitute Eq.9.11,

$$\begin{aligned}
&|\Lambda_i^{(m)}(t_{k+1}) - \Lambda_i(t_{k+1})| \qquad (9.12)\\
=&|\Lambda_i^{(m)}(t_k) - \Lambda_i(t_k) + h_m(\Lambda_i^{(m)})'(t_k) - h_m\Lambda_i'(t_k) + h_m e^{(m)}|\\
\leq&|\Lambda_i^{(m)}(t_k) - \Lambda_i(t_k)| + h_m|(\Lambda_i^{(m)})'(t_k) - \Lambda_i'(t_k)| + h_m|e^{(m)}|\\
\leq&|\Lambda_i^{(m)}(t_k) - \Lambda_i(t_k)| + h_m L_i|\Lambda_i^{(m)}(t_k) - \Lambda_i(t_k)| + h_m|e^{(m)}|\\
\leq&(1 + h_m L_i)|\Lambda_i^{(m)}(t_k) - \Lambda_i(t_k)| + h_m|e^{(m)}|
\end{aligned}$$

We denote the estimation error at time t_{k+1} as $W_{k+1} = \Lambda_i^{(m)}(t_{k+1}) - \Lambda_i(t_{k+1})$, accordingly,

$$\begin{aligned}
W_{k+1} \leq&(1 + h_m L_i)W_k + h_m|e^{(m)}|\\
\leq&\big(W_{k-1}(1 + h_m L_i) + h_m|e^{(m)}|\big)(1 + h_m L_i) + h_m|e^{(m)}|\\
&\vdots\\
\leq&W_0(1 + h_m L_i)^{k+1}\\
&+ h_m|e^{(m)}|\big(1 + (1 + h_m L_i) + \cdots + (1 + h_m L_i)^i\big)\\
\leq&W_0(1 + h_m L_i)^{k+1} + h_m|e^{(m)}|\frac{(1 + h_m L_i)^{k+1} - 1}{1 + h_m L_i - 1}\\
\leq&W_0(1 + h_m L_i)^{k+1} + \frac{|e^{(m)}|}{L_i}\big((1 + h_m L_i)^{k+1} - 1\big)\\
\leq&W_0(1 + h_m L_i)^{k+1} + \frac{|e^{(m)}|}{L_i}(1 + h_m L_i)^{k+1}\\
\leq&(1 + h_m L_i)^{k+1}\big(W_0 + \frac{|e^{(m)}|}{L_i}\big)\\
\leq&e^{h_m L_i(k+1)}\big(W_0 + \frac{|e^{(m)}|}{L_i}\big)
\end{aligned}$$

That is, $W_k \leq e^{h_m L_i k}\big(W_0 + \frac{|e^{(m)}|}{L_i}\big)$. Substituting $h_m I_m = t$ and $W_0 = 0$ into the above equation, we obtain the upper bound of the estimation error in the m^{th} iteration:

$$W_{I_m} \leq e^{L_i t}\frac{|e^{(m)}|}{L_i} = \frac{O(\tau)}{L_i}e^{L_i t}$$

That is,

$$|\Lambda_i(t) - \Lambda_i^{(m)}(t)| \leq \frac{O(\triangle t)}{L_i}e^{L_i t}$$

wherein $O(\triangle t)$ denotes the first-order truncation error. ∎

Proof of Lemma 7.2 (Sketch). During each iteration of inferring CHASSIS, we first run the parametric procedure. Evaluating the log-likelihood (updating the intensity function and its integral) requires $O(\max\{\mathcal{I}_m\} \times m)$ operations according to Eq. 9.7, and updating the parameters Θ costs

$O(M)$ operations according to the $14^{\text{th}} \sim 18^{\text{th}}$ lines of Algorithm 4. Therefore, per iteration computation cost of the parametric procedure is $O(M + \max\{\mathcal{I}_m\} \times m)$. ∎

Proof of Theorem 7.3 (Sketch). We use $\overline{\lambda_i}(s)$ to indicate the mean intensity conditioned on $\mathbf{X}_t$ over the time duration $[0, \bar{t} - t]$. Substituting the following Taylor approximation:

$$\lambda_i(t) \approx \mathcal{F}_i(\mu_i) + \mathcal{F}_i'(\mu_i) \sum_{j\in[M]} \sum_{t_{jl}<t} \alpha_{ij}(t)\phi_{ij}(t - t_{jl}) \quad (9.13)$$

into $\lambda_i(s)$, we obtain the following Volterra integral equation:

$$\begin{aligned}
\overline{\lambda_i}(s) &= \mathbb{E}[\lambda_i(s)|\mathcal{H}_t] \quad (9.14)\\
&= \mathbb{E}\left[\mathcal{F}_i(\mu_i) + \mathcal{F}_i'(\mu_i) \sum_{j\in[M]} \sum_{t_{jl}<s} \alpha_{ij}(t)\phi_{ij}(s - t_{jl})|\mathcal{H}_t\right]\\
&= \mathbb{E}\left[\mathcal{F}_i(\mu_i) + \mathcal{F}_i'(\mu_i) \sum_{j\in[M]} \alpha_{ij}(t) \int_0^s \phi_{ij}(s-\tau) dN_j(\tau)|\mathcal{H}_t\right]\\
&= \mathbb{E}\left[\mathcal{F}_i(\mu_i) + \mathcal{F}_i'(\mu_i) \sum_{j\in[M]} \alpha_{ij}(t) \int_0^s \phi_{ij}(s-\tau)\lambda_j(\tau) d\tau|\mathcal{H}_t\right]\\
&= \mathcal{F}_i(\mu_i) + \mathcal{F}_i'(\mu_i) \sum_{j\in[M]} \alpha_{ij}(t) \int_0^s \phi_{ij}(s-\tau)\mathbb{E}\left[\lambda_j(\tau)|\mathcal{H}_t\right] d\tau\\
&= \mathcal{F}_i(\mu_i) + \mathcal{F}_i'(\mu_i) \sum_{j\in[M]} \alpha_{ij}(t) \int_0^s \phi_{ij}(s-\tau)\overline{\lambda_j}(\tau) d\tau
\end{aligned}$$

The above equation could be solved numerically via the trapezoidal rule [48]. ∎

9.2 Additional Experiments

Comparison with other integration methods. We set two types of Hawkes processes referring to our semi-parametric inference approach (Section 7) as follows:

- L-HP: the linear Hawkes processes associated with the intensity in Eq.3.2 under our modified Euler integration method (denoted as LE-HP), Trapezoidal rule [18] (denoted as LT-HP), and Fourth Order Runge-Kutta method (denoted as LF-HP) respectively.
- E-HP: the exponential Hawkes processes associated with the intensity in Eq.3.2 under our modified Euler integration method (denoted as EE-HP), Trapezoidal rule (denoted as ET-HP), and Fourth Order Runge-Kutta method (denoted as EF-HP) respectively.

Initially, the base intensity μ is sampled from a uniform distribution over $[0, 0.01]$ for each dimension, and the coefficients $\{\alpha_{ij}\}_{i,j\in[M]}$ are generated from a uniform distribution on $[0, 0.1]$. We compare the performance of these three integration methods in terms of accuracy (LogLike) and convergence speed (computation time) on *SF* and *ST* datasets (see details in Section 8). For each dataset, we order all activities chronologically, and use the first 30%, 50%, 60%, 70%, 80% samples for training, respectively.

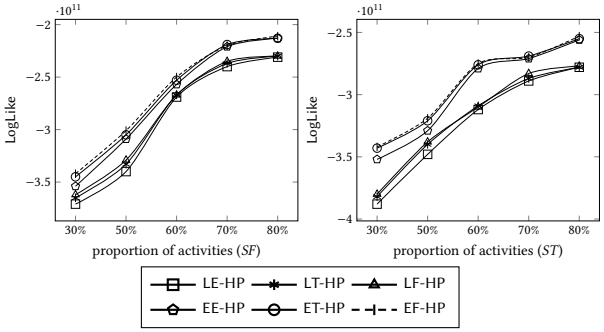


Figure 11. Accuracy of integration methods.

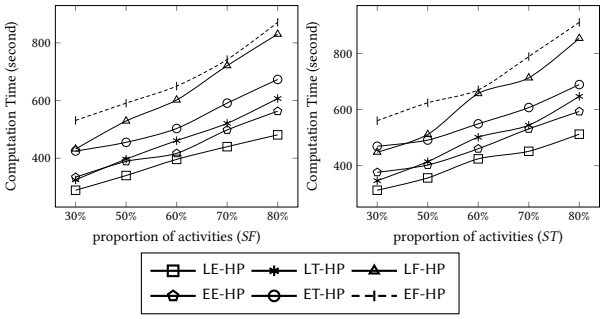


Figure 12. Convergence of integration methods.

Figure 11 shows the performance of different integration methods manifested through LogLike. Clearly, under the same Hawkes process (*i.e.,* the same form of intensity function in Eq.3.2), the three integration methods exhibit similar accuracy on abundant activities (*i.e.,* the time interval is long enough).

Figure 12 demonstrates the computation time under different integration methods. Under the same Hawkes process (*i.e.,* the same form of intensity function in Eq. 3.2), our modified Euler integration method shows superior convergence speed than the Trapezoidal rule and the Fourth Order Runge-Kutta method. That is, LE-HP (resp. EE-HP) is superior to LT-HP (resp. ET-HP) and LF-HP (resp. EF-HP). Accordingly, our modified Euler integration method shows the best efficiency. Note that in the numerical scheme, the Trapezoidal rule requires evaluating $\lambda_i(t)$ twice at each timestep and the Fourth Order Runge-Kutta method requires evaluating $\lambda_i(t)$ four times at each timestep. In contrast, the Euler integration only calculates $\lambda_i(t)$ once at each timestep.