A Component-Based Diffusion Model With Structural Diversity for Social Networks

Qing Bao, William K. Cheung, Yu Zhang, and Jiming Liu, *Fellow, IEEE*

***Abstract*—Diffusion on social networks refers to the process where opinions are spread via the connected nodes. Given a set of observed information cascades, one can infer the underlying diffusion process for social network analysis. The independent cascade model (IC model) is a widely adopted diffusion model where a node is assumed to be activated independently by any one of its neighbors. In reality, how a node will be activated also depends on how its neighbors are connected and activated. For instance, the opinions from the neighbors of the same social group are often similar and thus redundant. In this paper, we extend the IC model by considering that: 1) the information coming from the connected neighbors are similar and 2) the underlying redundancy can be modeled using a dynamic structural diversity measure of the neighbors. Our proposed model assumes each node to be activated independently by different communities (or components) of its parent nodes, each weighted by its effective size. An expectation maximization algorithm is derived to infer the model parameters. We compare the performance of the proposed model with the basic IC model and its variants using both synthetic data sets and a real-world data set containing news stories and Web blogs. Our empirical results show that incorporating the community structure of neighbors and the structural diversity measure into the diffusion model significantly improves the accuracy of the model, at the expense of only a reasonable increase in run-time.**

***Index Terms*—Diffusion networks, independent cascade model, social networks, structural diversity.**

I. Introduction

PEOPLE are often influenced by their friends to form opinions and views, resulting in information cascades. In social networks, the process is termed diffusion where information is spread via the connected nodes (with nodes modeling users and edges modeling their relationships). Given a set of observed information cascades, the underlying diffusion process can be inferred [1], [2] for different applications, including influence maximization [3]–[5], authoritative user identification [6], personalized recommendation [7], [8], etc.

The independent cascade (IC) model [1] and the linear threshold (LT) model [2] are two commonly used diffusion models for social networks. The IC model [1] assumes that a node can be activated independently by any one of its neighbors, while the LT model [2] assumes that whether a node will be activated depends on the aggregation of its neighbors' activations. In this paper, we extend the IC model by considering the structural diversity of node neighborhood to better model the diffusion processes in social networks. The basic IC model, since it was first proposed, has been extended in various ways, e.g., assuming node influence to decay over time as most people are more interested in recent news [9], [10], allowing the diffusion rate to be dynamic [11], [12], among others. To the best of our knowledge, no diffusion models have been proposed to take into account the effect of the structural diversity of neighbors on node activation, which is what we argue to be important.

Diffusion models are defined with the notion of neighborhood. The neighbors with direct connections (also called ties) to a node could exhibit different forms of influence depending on their connectivity in the social network. There have been studies on the effect of different local ties on the overall network properties. For instance, ties with different strength characterized by the amount of shared time, emotional intensity and so on have been found playing unique roles in a network [13]. The importance of weak ties serving as "local bridges" to introduce novel information in social networks has long been understood [13]. Related perspectives have recently been explored for online communication and social media analysis [14]–[16]. Onnela *et al*. [14] studied the roles of strong and weak ties in mobile communication networks and illustrated that random removal of weak ties could lead to the networks falling apart, no longer supporting the communication. Online social ties across heterogeneous networks have been studied in [16]. Also, the structure of neighbors has been considered as the resources they hold (also known as social capital) in [17]. Information provided by each neighbor when they communicate through their connectivity carries redundancy. In [15], it has been demonstrated that the number of connected components of the neighbors correlates well with the probability for a person joining social coalition.

Manuscript received September 28, 2015; revised January 4, 2016; accepted February 20, 2016. Date of publication March 21, 2016; date of current version March 15, 2017. This paper was supported in part by the General Research Fund through the Research Grants Council of Hong Kong Special Administrative Region under Grant HKBU210410, and in part by the Natural Science Foundation of China under Grant 61305071. This paper was recommended by Associate Editor J. Cao.

Q. Bao, W. K. Cheung, and J. Liu are with the Department of Computer Science, Hong Kong Baptist University, Hong Kong (e-mail: qingbao@comp.hkbu.edu.hk; william@comp.hkbu.edu.hk; jiming@comp.hkbu.edu.hk).

Y. Zhang is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, and also with the Institute of Research and Continuing Education, Hong Kong Baptist University, Shenzhen 518000, China (e-mail: yuzhang@comp.hkbu.edu.hk).

This paper has supplementary downloadable multimedia material available at http://ieeexplore.ieee.org provided by the authors. This includes a PDF file, which contains the derivation of the incremental method to compute the effective size [(9) in this paper]. The total size of the file is 58 KB.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2016.2537366

That is, it is not the number of friends influencing you that matters but the number of loosely coupled "groups" (or called nonredundant contacts in [17] and components in this paper).

We here propose a novel component-based IC model that considers the neighborhood structure of each node for modeling information redundancy during the diffusion process. In particular, a node will be activated independently by groups of parent nodes which are densely connected (called component in the sequel) instead of individual parent nodes. Also, we make use of different structural diversity measures for quantifying the redundancy of each component and then derive the corresponding model learning algorithm to infer the diffusion probabilities based on a set of observed cascades. The effectiveness of the proposed IC model is evaluated using both synthetic and real data sets. Note that the focus of this paper is to study the effect of incorporating neighbors' structural diversity into the diffusion network and the network structure is assumed to be known and static. The results of this paper can also be extended to the cases where the network structure is unknown [18] and/or contains dynamic ties [19], [20]. Also, we consider only static transmission rates and activations happening at discrete time steps.

The contributions of this paper are as follows.

1) We model information redundancy using the structural diversity of neighbors and propose a novel component-based diffusion model. To the best of our knowledge, we are the first group demonstrating the importance of considering the structural diversity of neighbors in diffusion modeling.
2) We adopt the notion of effective size in social science and propose a measure called dynamic effective size to allow the diffusion models to be more adaptive to dynamic behaviors.
3) We derive an expectation maximization (EM) algorithm for obtaining the ML estimates of the model parameters based on the observed cascades with a detailed analysis of its run-time.

The remainder of this paper is organized as follows. Section II presents related work on diffusion modeling. Detailed problem formulation and experimental results can be found in Sections III and IV. Section V concludes this paper and presents the future work.

II. Related Work

There has been a rapid growth in research on social network analysis [21], [22] over the past decade, and contagion in particular. Simple contagion (e.g., the IC model [1]) considers cases where only one source is sufficient for diffusion while complex contagion (e.g., the LT model [2]) is related to collective behaviors which require social affirmation from multiple sources [23]. There also exist different variants of the diffusion models in the literature. For instance, a node can be influenced by a linear combination of neighbors' influence, which is a generalization of the linear approximation for the IC model [6]. In [9] and [10], the influence of a parent node is assumed to decay over time after its activation in an exponential manner. Also, one can allow a node to be activated multiple times as it is reasonable for a motivated user to post several times on the same topic [24], [25]. In addition, a node can consider not only the influence of activations happened one time step before but also that of the earlier ones as a user could be motivated by revisiting earlier posts [9], [10]. Furthermore, one can allow the transmission rate to be dynamic in diffusion modeling [11], [12]. Asynchronous time models can allow the activations to occur in continuous time [26], [27]. Modeling the propagation of competing opinions [28], [29] has also been studied. Also, diffusion processes can evolve in time on temporal networks [20], [30].

This paper is also related to the study of social capital in social science. The structure of social contacts/neighbors in a network and the resources they each hold is generally defined as social capital [17]. Lacking edges among neighbors results in structural holes, which in turn benefit novel information. Notions like effective size and constraint have been defined as measures for structure holes in [17]. These form important concepts for formulating information redundancy, to be discussed in the next section.

III. Component-Based Diffusion Model

In this section, we put forward a diffusion based model for social networks with the consideration of the community structure of neighbors for each node. We regard the communities of the parents of a node as its components. Within a component, nodes are assumed to be frequently interacting and thus carry redundant information. So, instead of considering the independent influence of the neighboring nodes, we consider the independent influence of the neighboring components (as independent information sources [15], [17]). And for each component, we model the effective number of nodes by its effective size to further remove redundancy.

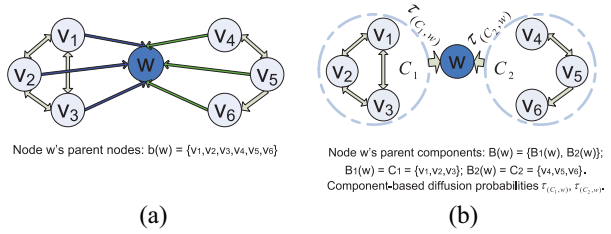
A. Preliminaries

We represent a given social network as a directed graph $G = (V, E)$ where V is the set of nodes and E is the set of edges. Let $e = (v, w)$ be an edge from node v to node w, and $f(v)$ and $b(v)$ be the sets of child nodes and parent nodes of node v, respectively, given as: $f(v) = \{w : (v, w) \in E\}$ and $b(v) = \{u : (u, v) \in E\}$. For each node w, we define the connected components of its parent nodes as its parent components $B(w) = \{B_i(w) : i = 1, \ldots, N_c(w)\}$ where $N_c(w)$ is the number of components for $B(w)$. And reversely, for a component c, we define the set of nodes having component c as one of its parent components as $F(c) = \{w : c \in B(w)\}$. Here, the parent components of node w are modeled as the detected communities in $b(w)$ using community detection algorithms (see [31]).

With the assumption that the component structure of the parents of each node is static, we define for each component-node pair $(c \in B(w), w)$ a component-based diffusion probability $\tau_{c,w}$ with $0 \leq \tau_{c,w} \leq 1$. Also, we allow a node to be activated multiple times. Some major notations defined in this paper are summarized in Table I. Fig. 1 illustrates a node w and its parents. The node w has a set of parent

TABLE I
NOTATIONS

SYMBOL	DESCRIPTION
$G = (V, E)$	A directed graph with node set V, edge set E
$f(v)$	The set of child nodes of node v
$b(v)$	The set of parent nodes of node v
$B(w)$	The set of parent components of node w
$N_c(w)$	The number of parent components of node w
$F(c)$	The set of nodes having parent component c
$\theta = \{\tau_{c,w}\}$	Component-based diffusion probability
$D_s = \{D_s(t)\}$	The s^{th} observed cascade
T_s	The end time for the cascade D_s
$D_s(t)$	The set of nodes activated at time t in s^{th} cascade
$C_s(w, t)$	The set of active parent components of w with respect to time step t in the s^{th} cascade
$G_c(t)$	The graph of activated parent nodes in c at time t with node set $N_c(t)$ and edge set $E_c(t)$
$F_a^{(s)}(c, w, t)$	The decay factor of c given w
$F_b^{(s)}(c, w, t)$	The structural diversity factor of c given w
$N_s(c, w, t)$	The effective count of activations in c
$T_c^{(s)}(w, t)$	The time when $F_b^{(s)}(c, w, t)$ is peaked
$L_s(w, t)$	The time difference between the current time t and the latest activation time of w up to time t
α	The decay parameter for $F_a^{(s)}(c, w, t)$



(a) (b)

Fig. 1. (a) Node-based versus (b) component-based diffusion.

nodes $b(w) = \{v_1, v_2, v_3, v_4, v_5, v_6\}$ [Fig. 1(a)]. The parent nodes form two components, i.e., $B(w) = \{B_1(w) = \{v_1, v_2, v_3\}, B_2(w) = \{v_4, v_5, v_6\}\}$ [Fig. 1(b)]. By denoting $B_1(w)$ as C_1 and $B_2(w)$ as C_2, the corresponding component-based diffusion probabilities are denoted as $\tau_{C_1,w}$ and $\tau_{C_2,w}$, respectively.

B. Problem Formulation

Let $D_s = \{D_s(0), D_s(1) \cdots D_s(T_s)\}$ be the sth observed information cascade, where $D_s(t)$ is the set of nodes activated at time step t and T_s is the end time of cascade D_s. In our proposed diffusion model, given the sth cascade and the current time step t, whether a node w will be activated at the time step $t + 1$ depends on whether its parent components $B(w)$ are active or not during the time interval $[t - L_s(w, t), t]$. Here $L_s(w, t)$ denote the time difference between the current time t and the latest activation time of w up to t. And a parent component is considered active during the interval if at least one of its nodes is activated during the interval. This implies that we are only interested in recent news and that the posts prior to our previous posting have little influence on our future posting behavior. We define $C_s(w, t) \subset B(w)$ to be the set of active parent components of w with respect to time step t in the sth cascade.

The diffusion process of a particular cascade proceeds as follows. Given the initial set of activated nodes in the sth cascade ($D_s(0)$), the parent components of each node are checked for being active or not as the time step proceeds. Based on the diffusion probabilities $\{\tau_{c,w}\}$ with $c \in C_s(w, t)$, some of their child nodes $\{w\}$ will be activated accordingly. The process proceeds until there are no more nodes being activated and thus the cascade stops. To infer the diffusion model, we adopt the Bayesian framework and obtain the model parameters by maximizing the likelihood of generating the observed cascades $\{D_s\}$ (to be discussed in Section III-C).

In this paper, we incorporate also the factors which can affect the degree of influence of a component activation into the diffusion model. We regard a parent component's degree of influence to be affected by: 1) the time at which the component is activated and 2) the dynamics of the structural properties of the activated nodes in the component. We define the two factors as $F_a^{(s)}(c, w, t)$ and $F_b^{(s)}(c, w, t)$. In the following, we first introduce the structural diversity factor $F_b^{(s)}(c, w, t)$, and then the decay factor $F_a^{(s)}(c, w, t)$ which is defined based on $F_b^{(s)}(c, w, t)$.

1) Structural Diversity Factor: We argue that a parent component is more influential if the associated nodes are sparsely connected, and thus less redundancy among them. In social networks, news items posted by closely linked websites are considered to carry redundant information. We formulate $F_b^{(s)}(c, w, t)$ as the effective size, a well-known measure of structure holes [17]. As explained by the theory of social capital, lacking edges among neighbors results in structural holes, which benefit novel information [17].

To compute $F_b^{(s)}(c, w, t)$, we first build a weighted undirected complete graph $G_c(t) = (N_c(t), E_c(t))$ for the set of activated parent nodes $N_c(t)$ of w in component $c \in C_s(w, t)$ before time step $t + 1$ where $E_c(t)$ defines the set of pairs of the activated nodes in component c. For each edge $e_{ij} \in E_c(t)$, we compute a weight h_{ij} to indicate the similarity of the node pair. Even though the activated nodes are not connected at a certain time, they may share information through common friends, which is also known as structural equivalence [17]. For example, in Fig. 1, although v_4 and v_6 are not connected, they could share information via v_5. We use SimRank [32][1] that considers node connectivity to calculate the similarity score associated to each node pair in component c. The score will then be within $[0, 1]$. To achieve run-time efficiency, we use the similarity scores obtained after the first iteration, which is essentially equal to a normalized version of co-citation [33]. Given $\{h_{ij}\}$, we further define the relative similarity $m_{jq} \in [0, 1]$ as

$$m_{jq} = \frac{h_{jq}}{\max_{k \in N_c(t)} h_{jk}}. \quad (1)$$

To sum up the influence of the nodes in $N_c(t)$ on its child node w, we use once again SimRank to first compute the similarity of node w and each node in $N_c(t)$. We then define

[1]The use of SimRank is by no means optimal and alternatives can be explored in future work.

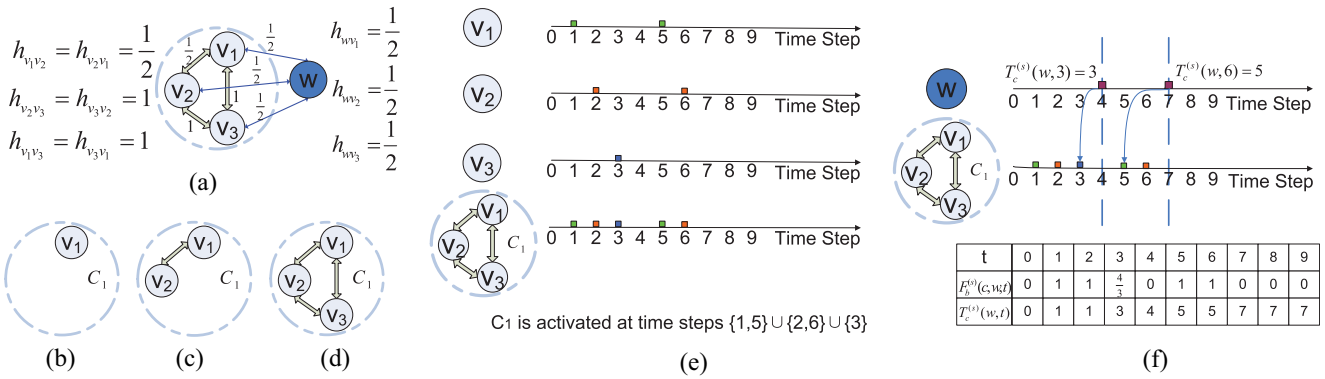


Fig. 2. (a)–(d) Calculation of the effective size. (e) Node activations. (f) Calculation of component activation time.

$p_{wq} \in [0, 1]$ as the portion of emphasis w will put on a parent node q, given as

$$p_{wq} = \frac{h_{wq} + \epsilon}{\sum_{j \in N_c(t)} (h_{wj} + \epsilon)} \tag{2}$$

with an additive smoothing parameter ϵ (set to 1E-12 in our experiments). Then, the effective size of $G_c(t)$ is given as

$$F_b^{(s)}(c, w, t) = \sum_{j \in N_c(t)} \left(1 - \sum_{q \in N_c(t) \setminus \{j\}} p_{wq}\, m_{jq}\right). \tag{3}$$

Given the formulation, $F_b^{(s)}(c, w, t)$ takes values within $[1, |N_c(t)|]$. In addition, $\sum_{q \in N_c(t) \setminus \{j\}} p_{wq}\, m_{jq}$ can be interpreted as the redundancy for parent node j. Note that if $\max_{k \in N_c(t)} h_{jk}$ equals 0, indicating h_{jq} equals 0 for all q, we assign m_{jq} to 0 since there is no information shared with any node q to cause redundancy.

It is interesting to note that if we compute the similarity score by assigning the weight h_{ij} to 1 given there exists a corresponding edge (e_{ij} or e_{ji}) in G and to 0 otherwise, (3) can be rewritten as

$$F_b^{(s)}(c, w, t) = |N_c(t)| - \frac{2\,|E_c(t)|}{|N_c(t)|} \tag{4}$$

which is equivalent to the normalization of modularity measure [34]. In the sequel, we refer to the use of (3) as adopting effective size, and (4) as adopting modularity. Meanwhile, if we consider all the parent nodes in a component instead of only the activated ones, $F_b^{(s)}(c, w, t)$ becomes a static measure, whereas the aforementioned measures are all dynamic by definition. The effectiveness of different versions of the structural diversity factor, namely dynamic effective size, static effective size, dynamic modularity, and static modularity will be evaluated and discussed in Section IV.

Example 1: The similarity scores of the node pairs in component C_1 are listed in Fig. 2(a). At $t = 1$ [Fig. 2(b)], only v_1 is activated. The inner sum in (3) has no items and thus $F_b^{(s)}(c, w, t)$ equals 1. At $t = 2$, v_2 is activated [Fig. 2(c)]. It is easy to see that $m_{v_1v_2} = m_{v_2v_1} = 1$ and $p_{v_1w} = p_{v_2w} = (1/2)$. Then, $F_b^{(s)}(c, w, t) = (1 - (1/2)) + (1 - (1/2)) = 1$. When v_3 is further activated at $t = 3$ [Fig. 2(d)], $m_{v_1v_2} = m_{v_2v_1} = (1/2)$, and for other pairs the values are 1. $p_{v_1w} = p_{v_2w} = p_{v_3w} = (1/3)$. Thus, $F_b^{(s)}(c, w, t) = (1-(1/3)\times(1/2) - (1/3)) + (1 - (1/3) \times (1/2) - (1/3)) + (1 - (1/3) - (1/3)) = (4/3)$.

2) Decay Factor: For the factor $F_a^{(s)}(c, w, t)$, we need to define a component activation start time so as to formulate the decay effect. We can postulate that a user will start paying attention to the posts in a parent component when the topic is first discussed or when it is frequently discussed among some nodes within the component. For the former, the definition is obvious. For the latter, we can compute $T_c^{(s)}(w, t)$ (peak time) which is the time t' when the value of $F_b(c, w, t')$ reaches maximum within the interval $[t - L_s(w, t), t]$. In case it reaches maximum at multiple time points, we take the earliest one. And in case there are no activations in the interval, we consider that the component c has no influence on node w, and $F_a^{(s)}(c, w, t)$ equals 0.

Given the activation start time of a parent component c to be $T_c^{(s)}(w, t)$, the component with the activation start time closer to the time t will be more influential on w at t. We adopt an exponential decay [9], [10] which gives

$$F_a^{(s)}(c, w, t) = 1 + e^{-\left(t - T_c^{(s)}(w,t)\right)/\alpha}. \tag{5}$$

The parameter α (also called the mean life time [10]) represents the expected time delay between an activation of a parent component and that of its child node. Note that $F_a^{(s)}(c, w, t)$ is formulated such that it is always larger than 1. Our preliminary experimental results show that adding the offset value 1 gives more stable performance.

Example 2: Fig. 2(e) and (f) illustrates the calculation of $T_c^{(s)}(w, t)$. As shown in Fig. 2(d), the component C_1 contains nodes $\{v_1, v_2, v_3\}$. Node v_1 is activated at time steps $\{1, 5\}$, v_2 at time steps $\{2, 6\}$, and v_3 at time step $\{3\}$. The activations of component $C_1 = \{v_1, v_2, v_3\}$ are considered to happen at the union of the activation time steps of the three nodes, i.e., $\{1, 5\} \cup \{2, 6\} \cup \{3\} = \{1, 2, 3, 5, 6\}$. Given that the activations of the component's child node w happen at time steps $\{4, 7\}$, as shown in Fig. 2(f), we obtain $L_s(w, 4 - 1) = 3$ (since there are no previous activations), and $L_s(w, 7 - 1) = 2$. Prior to time step 4, according to Example 1, at $t = 1$ and $t = 2$, the value of the effective size remains 1. At $t = 3$, v_3 is activated, and the effective size increases to (4/3). Thus, the

effective size of component C_1 reaches maximum within the interval $[4-1-L_s(w,4-1),4-1]=[0,3]$ at time step 3. Therefore, $T_c^{(s)}(w,3)=3$. For time step 7, the effective size of component C_1 reaches maximum within the interval $[7-1-L_s(w,7-1),7-1]=[4,6]$ when v_1 is activated, i.e., $T_c^{(s)}(w,6)=5$.

3) Overall Formulation: By combining the structure diversity and decay factors, the overall influence (and we call it effective count in the sequel) of the activation of a parent component c on node w can be modeled as

$$N_s(c,w,t)=F_a^{(s)}(c,w,t)F_b^{(s)}\Big(c,w,T_c^{(s)}(w,t)\Big). \tag{6}$$

The probability that the node w becomes active at time $t+1$ is given as

$$P_w^{(s)}(t+1)=1-\prod_{c\in C_s(w,t)}\big(1-\tau_{c,w}\big)^{N_s(c,w,t)}. \tag{7}$$

Given $D=\{D_s:s=1,\ldots,S\}$ as the set of independent information diffusion cascades, and $\boldsymbol{\theta}=\{\tau_{c,w}\}$ as the set of diffusion probabilities, the log-likelihood function with respect to $\boldsymbol{\theta}$ can be written as

$$\begin{aligned}L(\boldsymbol{\theta})&=\sum_{s=1}^{S}\ln P(D_s|\boldsymbol{\theta},D_s(0))\\&=\sum_{s=1}^{S}\sum_{t=0}^{T_s-1}\Bigg(\sum_{w\in D_s(t+1)}\ln P_w^{(s)}(t+1)\\&\qquad+\sum_{w\notin D_s(t+1)}\sum_{c\in C_s(w,t)}N_s(c,w,t)\ln\big(1-\tau_{c,w}\big)\Bigg)\end{aligned} \tag{8}$$

where $D_s(0)$ are the nodes which are activated initially as the original sources in the sth cascade. Since there could be multiple paths from multiple sources in a cascade, $D_s(0)$ could consist of more than one node.

Then, the remaining step is to estimate $\boldsymbol{\theta}=\{\tau_{c,w}\}$ so as to maximize (8).

C. Learning Algorithm

We first identify the parent components for each node in the social network using community detection algorithms [31], [35]–[37]. The "Clauset–Newman–Moore" (CNM) algorithm [31] is adopted for the community/component detection in most of our experiments. We also evaluate the use of "InfoMap" [35] as an alternative for comparison (see Section IV-F). We then compute the effective counts of component activations $N_s(c,w,t)$ for defining the likelihood function. An EM algorithm is derived to obtain the ML estimates of the model parameters based on the observed cascades. The framework of our learning algorithm is shown in Fig. 3.

1) Effective Counts of Component Activations: The detailed steps for calculating the effective counts of component activations $N_s(c,w,t)$ is summarized in Algorithm 1. In words, we first precompute the nodes in component c which could cause the activation of node w at time step t in the sth cascade and store them as $T_c(w,c,s,t)$ where all the nodes in each component are to be traversed. Then we compute $N_s(c,w,t)$ based on $T_c(w,c,s,t)$. This is to avoid the time-consuming enumeration of iterators in a for-loop for computing $N_s(c,w,t)$. Also, we precompute the sum of $N_s(c,w,t)$ for cases where w is

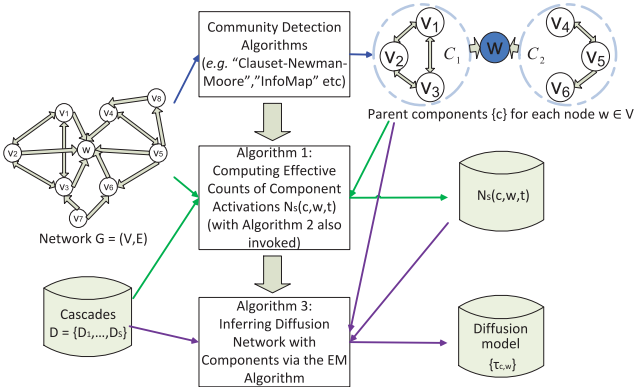


Fig. 3. Framework of our learning algorithm.

Algorithm 1 Computing Effective Counts of Component Activations

Input: network $G=(V,E)$, cascades $D=\{D_1,\ldots,D_S\}$
Output: effective counts of activations
for each parent component c of each node $w\in V$
at each time t in each cascade D_s, $N_s(c,w,t)$

global *max*, *prod*, *sum*
for all (c,w):$w\in V$ and $c\in B(w)$ **do**
  for all $i\in c$ **do**
    for all s:$i\in D_s$ **do**
      for all t:$w\in D_s(t+1)$ or $t=T_s-1$ **do**
        $T_c(w,c,s,t)\leftarrow T_c(w,c,s,t)$
        $\cup\{(i,min\{t'$:$i\in D_s(t')$ and $t'\in[t-L_s(w,t),t]\})\}$
      end for
    end for
  end for
  for all s:$\exists i\in c$:$i\in D_s$ **do**
    for all t:$w\in D_s(t+1)$ or $t=T_s-1$ **do**
      $Nodes\leftarrow Nil$ $F_b^{max}\leftarrow 0$ $T^{max}\leftarrow 0$
      SORT $(T_c(w,c,s,t),(i,t)\in T_c(w,c,s,t)$ by $t)$
      for all t':$\exists i$:$(i,t')\in T_c(w,c,s,t)$ **do**
        $NewNodes\leftarrow i$:$(i,t')\in T_c(w,c,s,t)$
        $F_b^{(s)}(c,w,t')\leftarrow$
        Calculating_the_Effective_Size($Nodes$, $NewNodes$)
        if $F_b^{(s)}(c,w,t')>max\{F_b^{(s)}(c,w,t''),$
          $t''\in[t-L_s(w,t),t')\}$ **then**
          $\sum_{t''\in(T^{max},t']}N_s(c,w,t'')\leftarrow$
          $\sum_{t''\in(T^{max},t']}(1+e^{-(t''-T^{max})/\alpha})F_b^{max}$
          $F_b^{max}\leftarrow F_b^{(s)}(c,w,t')$ $T^{max}\leftarrow t'$
        end if
        $Nodes\leftarrow Nodes\cup NewNodes$
      end for
      if $t\neq T_s-1$ **then**
        $F_a^{(s)}(c,w,t)\leftarrow 1+e^{-(t-T^{max})/\alpha}$
        $N_s(c,w,t)\leftarrow F_a^{(s)}(c,w,t)F_b^{max}$
      end if
      $\sum_{t'\in(T^{max},t]}N_s(c,w,t')\leftarrow$
      $\sum_{t'\in(T^{max},t]}(e^{-(t'-T^{max})/\alpha}+1)F_b^{max}$
    end for
  end for
end for

Algorithm 2 Calculating the Effective Size

Input: current nodes *Nodes*, the newly added nodes *NewNodes*
Output: $F_b^{(s)}(c, w, t')$

```
if Nodes = Nil then
    max ← 0    prod ← 0    sum(w) ← 0
end if
for all i ∈ NewNodes do
    for all j ∈ Nodes do
        max(i) ← max(h_ij, max(i))
        max(j) ← max(h_ij, max(j))
        prod(i) ← prod(i) + h_ij(h_jw + ε)
        prod(j) ← prod(j) + h_ij(h_iw + ε)
    end for
    Nodes ← Nodes ∪ i
    sum(w) ← sum(w) + h_iw + ε
end for
temp ← 0
for all i ∈ Nodes do
    if max(i) ≠ 0 then
        temp ← temp + prod(i)/max(i)
    end if
end for
F_b^(s)(c, w, t') ← |Nodes| − temp/sum(w)
```

not activated. $F_b^{(s)}(c, w, t)$ (3) is computed incrementally when there are new nodes to be added according to Algorithm 2. Given a set of newly added nodes, a naïve way to calculate $F_b^{(s)}(c, w, t)$ is to do it with two levels of summations, resulting in a quadratic runtime with respect to the number of nodes. Instead, it is not difficult to show that the formulation of the effective size can be rewritten as

$$F_b^{(s)}(c, w, t) = |N_c(t)| - \frac{\sum_{j \in N_c(t)} \frac{\text{prod}(j)}{\max(j)}}{\text{sum}(w)} \tag{9}$$

where $\text{prod}(j) = \sum_{q \in N_c(t) \setminus \{j\}} h_{jq}(h_{wq} + \epsilon)$, $\max(j) = \max_{k \in N_c(t)} h_{jk}$, and $\text{sum}(w) = \sum_{j \in N_c(t)} (h_{wj} + \epsilon)$. $\text{prod}(j)$, $\max(j)$ and $\text{sum}(w)$ can be updated via scanning the current set of nodes once when a new node j is added. The effective size is computed by traversing $\text{prod}(j)$ and $\max(j)$ of all the nodes in the second scan. Thus, updating $F_b^{(s)}(c, w, t)$ takes linear time.

2) Inferring Model Parameters: We make use of the EM [38] algorithm to infer the model parameters. We denote $Y_{c,w}^{(s)}(t)$ as the latent variable to indicate whether the activation of node w at time step t in the sth cascade is activated by w's parent component c. With reference to D_s, we represent the corresponding set of latent variables as $Y_s = \{Y_s(0), Y_s(1) \ldots Y_s(T_s)\}$ where $Y_s(t) = \{Y_{c,w}^{(s)}(t)\}$. We then derive the $\mathcal{Q}$-function and infer the model parameters via the EM algorithm which consists of an E-step and an M-step.

a) E-step: We take expectation of all possible assignments of Y which can explain the observed cascades.

Given a node $w \in D_s(t+1)$ and its parent component $c \in C_s(w, t)$, the probability of successful activation is $(1 - (1 - \tau_{c,w})^{N_s(c,w,t)})$ given $Y_{c,w}^{(s)}(t+1) = 1$. The probability of failing to activate, i.e., $Y_{c,w}^{(s)}(t+1) = 0$, is $(1 - \tau_{c,w})^{N_s(c,w,t)}$.

The $\mathcal{Q}$-function becomes

$$\begin{aligned}
&\mathcal{Q}\left(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}}\right) \\
&= \sum_{s=1}^{S} \sum_{t=0}^{T_s - 1} \Bigg(\sum_{w \notin D_s(t+1)} \sum_{c \in C_s(w,t)} N_s(c, w, t) \ln\left(1 - \tau_{c,w}\right) \\
&\quad + \sum_{w \in D_s(t+1)} \sum_{c \in C_s(w,t)} \Big(P\Big(Y_{c,w}^{(s)}(t+1) = 1\Big) \\
&\qquad \ln\Big(1 - \left(1 - \tau_{c,w}\right)^{N_s(c,w,t)}\Big) \\
&\quad + \Big(1 - P\Big(Y_{c,w}^{(s)}(t+1) = 1\Big)\Big) N_s(c, w, t) \\
&\qquad \ln\left(1 - \tau_{c,w}\right)\Big)\Bigg)
\end{aligned}$$

where the probability for the activated parent component c of node w to succeed in activating w at time step $t+1$ is calculated as

$$P\Big(Y_{c,w}^{(s)}(t+1) = 1\Big) = \frac{1 - \left(1 - \hat{\tau}_{c,w}\right)^{N_s(c,w,t)}}{\hat{P}_w^{(s)}(t+1)}$$

where $\hat{\tau}_{c,w}$ denotes the current estimate of $\tau_{c,w}$, and $\hat{P}_w^{(s)}(t+1)$ is computed according to (7).

b) M-step: We solve the optimality condition $\partial \mathcal{Q} / \partial \tau_{c,w} = 0$ for the new estimate of $\tau_{c,w}$.

We define $T_{c,w,s}^{+}$ ($T_{c,w,s}^{-}$) as the set of time steps $\{t\}$ in D_s where node w is (not) activated at t and at the same time its parent component c has been activated since $L_w^{(s)}(t)$. Also, we define the set of cascades where $T_{c,w,s}^{+}$ is not empty as $S_{c,w}^{+} = \{D_s : \exists t (c \in C_s(w, t) \wedge w \in D_s(t+1))\}$ and the set of cascades where $T_{c,w,s}^{-}$ is not empty as $S_{c,w}^{-} = \{D_s : \exists t (c \in C_s(w, t) \wedge w \notin D_s(t+1))\}$. Then

$$\begin{aligned}
\partial \mathcal{Q} / \partial \tau_{c,w} &= 0 \\
&\Rightarrow \sum_{s \in S_{c,w}^{+}} \sum_{t \in T_{c,w,s}^{+}} \Bigg(\frac{1 - \left(1 - \hat{\tau}_{c,w}\right)^{N_s(c,w,t-1)}}{\hat{P}_w^{(s)}(t)} \\
&\qquad N_s(c, w, t-1) \\
&\qquad \frac{1}{1 - \left(1 - \tau_{c,w}\right)^{N_s(c,w,t-1)}} \Bigg) \\
&= N_{c,w} = N_{c,w}^{+} + N_{c,w}^{-} \\
N_{c,w}^{+} &= \sum_{s \in S_{c,w}^{+}} \sum_{t \in T_{c,w,s}^{+}} N_s(c, w, t-1) \\
N_{c,w}^{-} &= \sum_{s \in S_{c,w}^{-}} \sum_{t \in T_{c,w,s}^{-}} N_s(c, w, t-1).
\end{aligned}$$

As the function

$$\begin{aligned}
f\left(\tau_{c,w}\right) = \sum_{s \in S_{c,w}^{+}} \sum_{t \in T_{c,w,s}^{+}} \Bigg(&\frac{1 - \left(1 - \hat{\tau}_{c,w}\right)^{N_s(c,w,t-1)}}{\hat{P}_w^{(s)}(t)} \\
&N_s(c, w, t-1) \frac{1}{1 - \left(1 - \tau_{c,w}\right)^{N_s(c,w,t-1)}} \Bigg) - N_{c,w}
\end{aligned}$$

Algorithm 3 Inferring Diffusion Network With Components

Input: network $G = (V, E)$, cascades $D = \{D_1, \ldots, D_S\}$, parent components $\{c\}$ for each node $w \in V$
Output: component-based diffusion probabilities $\boldsymbol{\theta} = \{\tau_{c,w}\}$

1. Assign initial values to $\hat{\boldsymbol{\theta}} = \{\hat{\tau}_{c,w}\}$
2. **for all** (c, w) pairs **do**
3. $N^+_{c,w} \leftarrow \sum_{s \in S^+_{c,w}} \sum_{t \in T^+_{c,w,s}} N_s(c, w, t-1)$
4. $N^-_{c,w} \leftarrow \sum_{s \in S^-_{c,w}} \sum_{t \in T^-_{c,w,s}} N_s(c, w, t-1)$
5. **if** $N^+_{c,w} = 0$ **and** $N^-_{c,w} \neq 0$ **then**
6. $\tau_{c,w} \leftarrow 0$
7. **end if** //special cases for diffusion probabilities
8. **end for**
9. **while** not convergence **do**
10. E-step:
11. **for all** $\hat{P}^{(s)}_w(t)$ **do**
12. $\hat{P}^{(s)}_w(t) \leftarrow 1 - \prod_{c \in C_s(w,t-1)} (1 - \hat{\tau}_{c,w})^{N_s(c,w,t-1)}$
13. **end for**
14. M-step:
15. **for all** (c, w):$S^+_{c,w} \neq \emptyset$ **do**
16. calculate $\tau_{c,w}$ using the bisection method for function
$$\sum_{s \in S^+_{c,w}} \sum_{t \in T^+_{c,w,s}} \left(\frac{1-(1-\hat{\tau}_{c,w})^{N_s(c,w,t-1)}}{\hat{P}^{(s)}_w(t)} N_s(c, w, t-1) \frac{1}{1-(1-\tau_{c,w})^{N_s(c,w,t-1)}} \right) - N_{c,w} = 0.$$
17. **end for**
18. $\hat{\boldsymbol{\theta}} \leftarrow \boldsymbol{\theta}$
19. **end while**

is monotonic, we use the bisection method to get the solution of $f(\tau_{c,w}) = 0$. For our case, the starting interval to solve for $\tau_{c,w}$ is set to $[0, 1]$ satisfying the condition $f(0)f(1) \leq 0$ for the bisection method to work.

When $N^+_{c,w} = 0$, namely $S^+_{c,w} = \emptyset$, then

$$\mathcal{Q}\big(\tau_{c,w} | \hat{\tau}_{c,w}\big) = \sum_{s \in S^-_{c,w}} \sum_{t \in T^-_{c,w,s}} N_s(c, w, t) \ln\big(1 - \tau_{c,w}\big) + \text{const}$$

where const stands for terms without $\tau_{c,w}$ included. The function becomes monotonically decreasing, and the maximum value is reached when $\tau_{c,w}$ is set to the minimum possible value, i.e., 0.

The E-step and M-step repeat until convergence. The detailed steps are summarized in Algorithm 3.

3) Computational Complexity: Implementing the learning algorithm involves three main steps: 1) load the network data, the per-node neighbors' community structure and the cascades related data; 2) precompute the effective counts of components (preprocessing); and 3) carry out the EM iterations.

For step 1), the cost for loading the network data is $O(|V| + |E|)$. For loading the per-node neighbors' community structure, it includes the similarity scores for all the node pairs in each parent component (needed for computing the effective size). Given that the number of nodes in a parent component c of a node w is $n(w, c)$, the total number of similarity scores to be computed will be $\sum_{w \in V} \sum_{c \in B(w)} n(w, c)^2$. But, since for each node w, the cost for traversing the $n(w, c)$ nodes for all $c \in B(w)$ is equivalent to that of visiting all its parent nodes, $\sum_{w \in V} \sum_{c \in B(w)} n(w, c)$ essentially gives $|E|$. By denoting $I_{\max}$ to be the maximum indegree of the network, it is easy to see that the worst case complexity for loading the similarity scores is

$$\sum_{w \in V} \sum_{c \in B(w)} n(w, c)^2 \leq I_{\max} \sum_{w \in V} \sum_{c \in B(w)} n(w, c)$$

and thus $O(I_{\max} \times |E|)$. Regarding the cascades information, the worst case complexity is $O(S \times T)$ where T denotes the maximum length of a cascade record. Thus, the overall complexity is $O(|V| + I_{\max} \times |E| + S \times T)$.

For step 2), the first major preprocessing task is to compute $T_c(w, c, s, t)$ (lines 2–9 in Algorithm 1). The worst case complexity is essentially that of computing $T_c(w, c, s, t)$, that is, $\sum_{w \in V} \sum_{c \in B(w)} n(w, c) \times S \times T$, which gives $O(S \times T \times |E|)$. The second task is to compute the effective size for activated nodes in a component (lines 10–30). The node activations in a component c [stored in $T_c(w, c, s, t)$] are added incrementally and then the value of $F^{(s)}_b(c, w, t')$ is updated accordingly. The complexity for such an update is $O(n(w, c))$ (lines 7–12 in Algorithm 2). Thus, the overall complexity of adding all the nodes (lines 14–16) for updating $F^{(s)}_b(c, w, t')$ is $O(n(w, c)^2)$. Then, the worst case complexity becomes

$$\sum_{w \in V} \sum_{c \in B(w)} S \times T \times n(w, c)^2 \leq I_{\max} \sum_{w \in V} \sum_{c \in B(w)} S \times T \times n(w, c)$$

which gives $O(I_{\max} \times S \times T \times |E|)$. The overall complexity for the preprocessing step (i.e., Algorithm 1) is thus $O(I_{\max} \times S \times T \times |E|)$.

For the EM algorithm (step 3) as shown in Algorithm 3, we first calculate $N^+_{c,w}$ and $N^-_{c,w}$ (lines 2–8) and the corresponding complexity is

$$\sum_{w \in V} \sum_{c \in B(w)} S \times T \leq \sum_{w \in V} \sum_{c \in B(w)} n(w, c) \times S \times T$$

and thus $O(S \times T \times |E|)$. The main part of the EM algorithm corresponds to lines 9–19. In each iteration, the bisection method (line 16) is the most costly step with the complexity of $\sum_{w \in V} \sum_{c \in B(w)} S \times T$ which gives $O(S \times T \times |E|)$. By denoting the number of iterations in the bisection method as k, then the complexity for each EM iteration becomes $O(k \times S \times T \times |E|)$.

IV. Experiments

We evaluate the proposed model using both synthetic and real data sets. We show that the component-based diffusion model is more accurate in modeling diffusion when compared with the node-based ones. All the experiments are conducted on a machine with a 2.67 GHz 4-core CPU and 32 GB RAM running Linux. The algorithms are developed using C++. In the following, we first present the experimental settings for conducting the performance evaluation.

A. Experimental Settings

We first implement a basic node-based IC model which extends the original IC model by considering the influence of all the parent nodes activated after the child node's latest activation instead of only those just activated at the previous time step. We use it as the baseline reference for evaluation.

The main reason of using this modified IC model is to make sure that the comparison is only based on whether the component structure is adopted or not but not other modeling aspects.

We then implement the component-based model with different extensions by adding the structure diversity factor and the decay factor into the model. In particular, we have tested the following combinations.

1) *ICM:* The basic IC model.
2) *ICM-DK:* The IC model with the decay factor.
3) *COMP:* The component-based IC model without the decay factor.
4) *COMP-DK:* The component-based IC model with the decay factor.

ICM and ICM-DK are essentially node-based, while the other two are component-based. For ICM and ICM-DK, the number of parameters for each node equals the number of its parent nodes, while for COMP and COMP-DK, the number is reduced to that of its parent components. And for both COMP and COMP-DK, we adopt the dynamic effective size to define the structural diversity factor.

To contrast the effectiveness of adopting different structural diversity factors as mentioned in Section III-B1 as well as different ways to define the time for a component to be considered activated, we implement the following variants of the component-based model.

1) *COMP(1st):* The component-based model without considering structural diversity. The time of the first node activation in the component is considered as the time of the component activation.
2) *COMP_SMod(1st):* The static modularity is adopted as the structural diversity factor. The time of the first node activation is considered as the time of the component activation.
3) *COMP_SEffSz(1st):* The static effective size is adopted as the structural diversity factor. The time of the first node activation is considered as the time of the component activation.
4) *COMP_DMod(Max):* The dynamic modularity is adopted as the structural diversity factor. The time when the dynamic modularity value reaches maximum is considered as the time of the component activation.
5) *COMP_DEffSz(Max):* The dynamic effective size is adopted as the structural diversity factor. The time when the dynamic effective size reaches maximum is considered as the time of the component activation.

For all the experiments, we set the initial values of $\hat{\boldsymbol{\theta}} = \{\hat{\tau}_{c,w}\}$ to be within $[0, 0.1]$ as the diffusion probabilities in real cases are small (e.g., with a mean value of 0.04 and standard deviation of 0.07 in [39]). We test different initializations and report the best results to get rid of the local minimum problem, though the variations are found to be only within 0.005 for almost all the runs. For detecting the community structure, we consider nodes within two hops (instead of just the direct neighbors) to enhance the detection accuracy. For the decay factor α, we try different values and find setting α to 100 works fine for all the models.

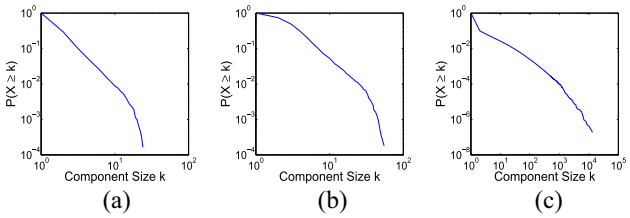


Fig. 4. Complementary cumulative distributions of parent component size in (a) and (b) synthetic networks with 5000 and 10 000 edges, respectively, and the (c) real network (MemeTracker).

B. Performance Evaluation

Given that the ground truths of the diffusion models are unknown, we adopt perplexity as the evaluation metric. The perplexity over the observed cascades is defined as

$$\text{Perplexity} = \frac{-\sum_{s=1}^{S} \ln P(D_s)}{W} \tag{10}$$

where $P(D_s)$ is the probability to generate the sth cascade, and the normalization term W is the number of activations due to the influence of the corresponding nodes' parents. A smaller perplexity value indicates the inferred model to be more probable. Fivefold cross-validation is adopted for all the experiments.

In addition, the simulation approach can also be used for the evaluation [40]. Information cascades can be generated using different diffusion models given the same set of initial node activations for each cascade in the test set to first estimate empirically the probabilities of different nodes being activated afterwards. Then, the nodes are ranked accordingly and the percentage of the top K nodes also found in the test set can be computed. The metric is commonly called precision at K, denoted as $P@K$. Again, fivefold cross-validation is adopted.

C. Experiments on Synthetic Data

We generate synthetic cascades based on the component-based IC model with the dynamic effective size adopted for the structural diversity factor. We anticipate that the inferred model with the same assumption for cascade generation should perform the best.

1) Experimental Setup: We first generate two scale-free networks of 1000 nodes using the SNAP platform [41] as real networks are mostly scale-free. One network is generated with 5000 edges and the other with 10 000 edges. For each network, 100 cascades are generated based on the proposed component-based model where the decay factor α is set to 100 and the diffusion probabilities are randomly assigned. Note that the network with 10 000 edges is denser and thus there are more activations in the cascades, providing more data for model training. Fig. 4(a) and (b) shows the complementary cumulative distributions of the size of the parent components k (i.e., the fraction of parent components that have nodes greater than or equal to k) in the two synthetic networks. The long tail distribution indicates the presence of parent components with a wide range of sizes.

2) Generative Ability: We apply ICM, ICM-DK, COMP, and COMP-DK to the synthetic networks. As mentioned

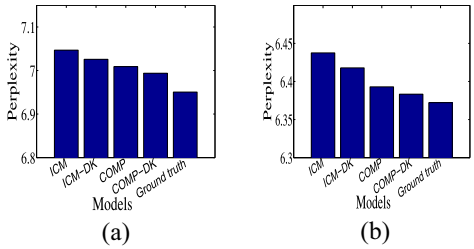


Fig. 5. Performance comparison on synthetic data for networks with (a) 5000 edges and (b) 10 000 edges.

TABLE II
PERFORMANCE COMPARISON IN TERMS OF P@K BASED ON THE SYNTHETIC DATA. THE BEST RESULTS ARE PRINTED IN BOLDFACE

Model	5,000 edges			10,000 edges		
	P@10	P@50	P@100	P@10	P@50	P@100
ICM	0.487	0.487	0.487	0.887	0.887	0.887
ICM-DK	0.701	0.587	0.567	**0.964**	0.954	0.941
COMP	0.620	0.605	0.596	0.891	0.891	0.891
COMP-DK	**0.836**	**0.760**	**0.705**	0.958	**0.956**	**0.954**

TABLE III
PERFORMANCE COMPARISON AMONG THE COMPONENT-BASED MODELS IN TERMS OF PERPLEXITY BASED ON THE SYNTHETIC DATA. THE BEST RESULTS ARE PRINTED IN BOLDFACE

Settings	5,000 edges		10,000 edges	
	without decay	with decay	without decay	with decay
COMP(1st) (baseline)	7.013	7.002	6.415	6.430
COMP_SMod(1st)	7.014	7.003	6.415	6.431
COMP_SEffSz(1st)	7.014	7.003	6.415	6.431
COMP_DMod(Max)	7.022	7.005	6.419	6.402
COMP_DEffSz(Max)	**7.009**	**6.994**	**6.393**	**6.383**

in Section IV-A, we adopt the dynamic effective size (COMP_DEffSz(Max)) for both COMP and COMP-DK to define the structural diversity factor. The performance comparison results in terms of perplexity and $P@K$ are shown in Fig. 5 and Table II, respectively. According to Fig. 5, we observe that all the models perform better for the network with 10 000 edges when compared with that with 5000 edges due to more training data. Also, while adding the decay factor can result in a perplexity decrease of 0.02 for the two networks, adding the structure diversity factor achieves more significant improvement with a perplexity decrease of 0.04. Combining both, the performance further improves by a perplexity decrease of 0.02 and 0.01, respectively, for the two networks.

In addition, Table II shows the performance measured in terms of $P@K$ for $K = \{10, 50, 100\}$. The performance ranking among the models remains more or less the same given different values of K. COMP and COMP-DK apparently outperform ICM and ICM-DK, especially when the data is sparse (5000 edges).

Then, we follow the experiment protocols described in Section IV-A. Table III shows the performance of different variants of the component-based model. Among them, COMP_DEffSz(Max) achieves the best performance. Again, as anticipated, for the network with 10 000 edges, more apparent improvement is achieved.

By contrasting the performance of COMP_SEffSz(1st) versus COMP_SMod(1st) and COMP_DEffSz(Max) versus COMP_DMod(Max), we see that the use of the effective size gives better results than using modularity. In addition, by contrasting the performance of COMP_DEffSz(Max) versus COMP_SEffSz(1st) and COMP_DMod(Max) versus COMP_SMod(1st), the dynamic structural diversity measures are found better when compared with the static counterparts, except for COMP_DMod(Max) applied to the network with 5000 edges. For the decay factor, as shown in Table III, its inclusion improves the performance in most cases, except for the models with static structural diversity measures for the network with 10 000 edges.

D. Experiments on Real Data

To validate if component-based diffusion indeed happens in online social networks, we apply the proposed model to a real social network data set.

1) Data Set: We use the MemeTracker [42] data set that contains: 1) the link structure of websites with news articles and blog posts and 2) the corresponding information cascades. It covers a period of nine months from August 1, 2008 to April 30, 2009. A website A is assumed to have influence on a website B if a post in B has mentioned a post in A. Then, there will be an edge from node A to node B. The data set contains $4m$ nodes and $13m$ edges.

To investigate if it is common to have parent components with more than one node in the data set, we plot the corresponding complementary cumulative distribution as shown in Fig. 4(c) and observe that it follows the power law. Thus, having parent components with more than one node is highly probable, which hints modeling the structural diversity of node neighborhood is meaningful for the MemeTracker data set. As an illustrated example, we extract from the data set the parent nodes of the website "ksat.com," a local news website in San Antonio. The parent nodes form several communities which are found to be corresponding to: 1) general news ("news.bbc.co.uk," "cnn.com"); 2) business news ("economist.com," "forbes.com"); and 3) sports ("sports.espn.go.com"), and so on. It is not difficult to interpret that each group of the websites essentially forms an independent information source.

For the cascades [18], each is defined based on a frequently mentioned phrase or its variants in the posts. For each cascade, the time steps and the corresponding websites mentioning the phrase or its variants are recorded. The data set contains 71 568 cascades.

2) Generative Ability: We apply ICM, ICM-DK, COMP, and COMP-DK to the MemeTracker data set and the results are shown in Fig. 6(a). For COMP and COMP-DK, the dynamic effective size is used to measure structure diversity. Both component-based diffusion models give significantly lower perplexity values when compared with the node-based counterparts. When $\alpha = 100$, adding the decay factor improves the ICM with a perplexity decrease of 2.68. This indicates that the decay of influence is an important factor governing the diffusion. By incorporating the component-based formulation,

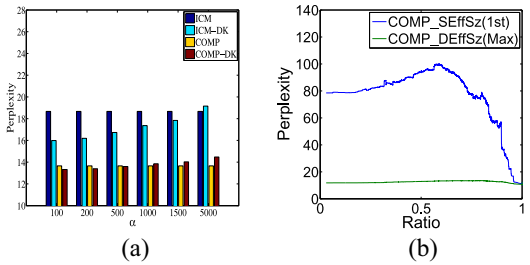


Fig. 6. (a) Performance comparison between node-based and component-based diffusion models given different values of α (MemeTracker). (b) Effect on the model accuracy given different ratios of nodes activated within an activated component (MemeTracker).

TABLE IV
PERFORMANCE COMPARISON OF VARIANTS OF THE COMPONENT-BASED MODEL BASED ON THE MEMETRACKER DATA SET. THE BEST RESULTS ARE PRINTED IN BOLDFACE

Settings	without decay	with decay
COMP(1st) (baseline)	13.729	13.438
COMP_SMod(1st)	121.705	127.475
COMP_SEffSz(1st)	101.259	106.006
COMP_DMod(Max)	13.765	13.442
COMP_DEffSz(Max)	**13.662**	**13.326**

the decrease in perplexity can reach 5.01. This indicates the validity of the proposed component-based diffusion models for social networks. Combining both factors, the performance further improves by an additional drop of 0.34 in perplexity.

Also, we compare the performance of the models inferred with the value of the decay coefficient α ranging from three days ($\alpha = 100$) up to five months ($\alpha = 5000$). Referring to Fig. 6(a), the perplexity values of ICM and COMP remain unchanged as they do not consider the decay factor at all. For ICM-DK and COMP-DK, the performance decreases as the value of α increases. This indicates that the influence decay should not be too slow. For instance, readers most likely do not pay attention to the posts appearing a few months ago.

Table IV shows the performance comparison of the models given different combinations of the structure diversity and decay factors. The results are consistent with those based on the synthetic data. Without the decay factor, COMP_DEffSz(Max) achieves the best performance (perplexity = 13.662). With the decay factor incorporated, COMP_DEffSz-DK(Max) achieves the best performance (perplexity = 13.326). In general, the settings with the effective size incorporated achieve better performance when compared with those using modularity, as shown in Table IV (COMP_SEffSz(1st) versus COMP_SMod(1st); COMP_DEffSz(Max) versus COMP_DMod(Max)). However, we observe that COMP_SEffSz(1st) and COMP_SMod(1st) perform extremely bad compared to their dynamic counterparts COMP_DEffSz(Max) and COMP_DMod(Max), which, however, is not observed when the synthetic data is used. To explain that, we further compared the static and dynamic models by referring to nodes with only selected parent components. In particular, for each node, we compute the ratio of the number of activated parent nodes to the total number of nodes in the activated parent components. We then set a lower bound on the ratio, and select different subsets of nodes for computing the corresponding perplexity values. When the lower bound is set to zero, it is equivalent to selecting all the nodes. When the lower bound is larger than zero, we start filtering out nodes with their activated parent components having a certain degree of their nodes not activated. When the ratio reaches one, only the nodes with their parent components containing only activated parents are selected. For such a case, the static and dynamic formulations should behave exactly the same. Fig. 6(b) shows the changes of the perplexity values of COMP_SEffSz(1st) and COMP_DEffSz(Max) as the lower bound on the ratio increases from zero to one. As anticipated, we observe that COMP_SEffSz(1st) and COMP_DEffSz(Max) give the same performance when the ratio lower bound is one. As the ratio lower bound is less than one, we find a substantial rise in perplexity for COMP_SEffSz(1st) while COMP_DEffSz(Max) still maintains a low perplexity value.

E. Run-Time

The run-time for: 1) loading the network and the cascades related information; 2) preprocessing the cascades; and 3) running the EM algorithm for both synthetic and real networks are shown in Fig. 7. In particular, we compare the run-time performance of ICM, ICM-DK, COMP_DEffSz(Max) (labeled as COMP), COMP_DEffSz(Max)-DK (labeled as COMP-DK), COMP(1st), and COMP-DK(1st). COMP_DEffSz(Max) and COMP_DEffSz(Max)-DK incur longer time for loading information as the similarity scores for all the node pairs in each parent component are involved.

Regarding the time for preprocessing cascades, the component-based models consume slightly more time as computing the effective counts of component activations involves aggregation of nodes' activations into the parent components' [Algorithm 1 (lines 3–8) with the complexity of $O(S \times T \times |E|)$ as discussed in Section III-C3]. Among them, COMP_DEffSz(Max) and COMP_DEffSz(Max)-DK take longer time mainly due to the computation of the effective size [Algorithm 1 (lines 14–22) with the complexity of $O(I_{\max} \times S \times T \times |E|)$].

For the run-time of the EM algorithm (Algorithm 3), ICM-DK takes significantly long time than ICM as adding the decay factor requires the use of the bisection method for estimating the corresponding parameters [the complexity becomes $O(k \times S \times T \times |E|)$ instead of $O(S \times T \times |E|)$ as presented in Section III-C3]. The run-time needed by the component-based models drops significantly as all the computations are basically component-based instead of node-based.

F. Sensitivity to Component Identification Methods

The key contribution of our paper is to demonstrate the importance of introducing the component-based notion in the diffusion modeling. Various algorithms for community detection can be utilized to detect parent components. In this section, we evaluate the sensitivity of the proposed component-based IC model given two different community detection algorithms, namely the "CNM" [31] and the InfoMap [35] algorithms. The results are presented in Table V. We find that

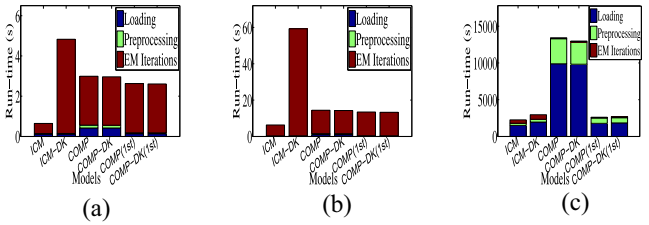


Fig. 7. Comparison of run-time for loading the network and the cascades related information, preprocessing the cascades, and running the EM algorithm on (a) and (b) synthetic networks with 5000 and 10 000 edges and (c) real data.

TABLE V
PERFORMANCE COMPARISON BASED ON DIFFERENT ALGORITHMS USED FOR THE COMMUNITY DETECTION STEP

Settings	without decay		with decay	
	CNM	InfoMap	CNM	InfoMap
COMP(1st)	13.729	14.414	13.438	14.053
COMP_DMod(Max)	13.765	14.464	13.442	14.109
COMP_DEffSz(Max)	13.662	14.360	13.326	13.990

the models inferred with CNM and InfoMap used in the community detection step give similar modeling accuracy, with the former one performs slightly better than the latter.

V. CONCLUSION

In this paper, we proposed a component-based IC model which incorporates the community structure of the node neighbors to model information diffusion. We adopted the effective size—a structural metric well-known in social science and extended it to a dynamic version for characterizing the influence of an activated parent component. An EM algorithm was derived for training the component-based IC model. With the proposed model, we obtained significant improvement on model accuracy at the expense of reasonable increase in run-time.

This paper has some limitations. Unlike some related work where the network structure is unknown [18], we assume that the network structure is known. And for simplicity, we assume that the diffusion rate is static and topic-independent, and that activations only occur at discrete time steps. Also, we assume that the cascade information obtained from the MemeTracker data set is correct and complete. For future work, the above assumptions can be further relaxed. In addition, other facets of structural properties besides redundancy can be considered to enhance the model accuracy. For instance, the hierarchical structure of the neighborhood can be explored for its importance to determine how influential a component activation should be. The proposed component-based IC model can also be applied to other network analysis tasks, e.g., influence maximization.

REFERENCES

[1] J. Goldenberg, B. Libai, and E. Muller, "Using complex systems analysis to advance marketing theory development," *Acad. Mark. Sci. Rev.*, vol. 9, no. 3, pp. 1–18, 2001.

[2] D. Kempe, J. Kleinberg, and E. Tardos, "Influential nodes in a diffusion model for social networks," in *Proc. 32nd Int. Conf. Autom. Lang. Program.*, Lisbon, Portugal, 2005, pp. 1127–1138.

[3] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Disc. Data Mining*, Washington, DC, USA, 2003, pp. 137–146.

[4] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Disc. Data Mining*, Paris, France, 2009, pp. 199–208.

[5] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan, "A data-based approach to social influence maximization," *Proc. VLDB Endowment*, vol. 5, no. 1, pp. 73–84, 2011.

[6] B. Xiang *et al.*, "PageRank with priors: An influence propagation perspective," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, Beijing, China, 2013, pp. 2740–2746.

[7] X. Song, Y. Chi, K. Hino, and B. L. Tseng, "Information flow modeling based on diffusion rate for prediction and ranking," in *Proc. 16th Int. Conf. World Wide Web*, Banff, AB, Canada, 2007, pp. 191–200.

[8] X. Song, B. L. Tseng, C.-Y. Lin, and M.-T. Sun, "Personalized recommendation driven by information flow," in *Proc. 29th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Seattle, WA, USA, 2006, pp. 509–516.

[9] W. Lee, J. Kim, and H. Yu, "CT-IC: Continuously activated and time-restricted independent cascade model for viral marketing," in *Proc. 12th Int. Conf. Data Mining*, Brussels, Belgium, 2012, pp. 960–965.

[10] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan, "Learning influence probabilities in social networks," in *Proc. 3rd ACM Int. Conf. Web Search Data Mining*, New York, NY, USA, 2010, pp. 241–250.

[11] M. Gomez-Rodriguez, D. Balduzzi, and B. Schölkopf, "Uncovering the temporal dynamics of diffusion networks," in *Proc. 28th Int. Conf. Mach. Learn.*, Bellevue, WA, USA, 2011, pp. 561–568.

[12] M. Gomez-Rodriguez and B. Schölkopf, "Influence maximization in continuous time diffusion networks," in *Proc. 29th Int. Conf. Mach. Learn.*, Edinburgh, U.K., 2012, pp. 313–320.

[13] M. S. Granovetter, "The strength of weak ties," *Amer. J. Sociol.*, vol. 78, no. 6, pp. 1360–1380, 1973.

[14] J.-P. Onnela *et al.*, "Structure and tie strengths in mobile communication networks," *Proc. Nat. Acad. Sci.*, vol. 104, no. 18, pp. 7332–7336, 2007.

[15] J. Ugander, L. Backstrom, C. Marlow, and J. Kleinberg, "Structural diversity in social contagion," *Proc. Nat. Acad. Sci.*, vol. 109, no. 16, pp. 5962–5966, 2012.

[16] J. Tang, T. Lou, and J. Kleinberg, "Inferring social ties across heterogenous networks," in *Proc. 15th ACM Int. Conf. Web Search Data Mining*, Seattle, WA, USA, 2012, pp. 743–752.

[17] R. S. Burt, *Structural Holes: The Social Structure of Competition*. Cambridge, MA, USA: Harvard Univ. Press, 1992.

[18] M. G. Rodriguez, J. Leskovec, and A. Krause, "Inferring networks of diffusion and influence," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Disc. Data Mining*, Washington, DC, USA, 2010, pp. 1019–1028.

[19] Z. Bu, Z. Wu, J. Cao, and Y. Jiang, "Local community mining on distributed and dynamic networks from a multi-agent perspective," *IEEE Trans. Cybern.* [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/26087512

[20] H. Habiba, "Critical individuals in dynamic population networks," Ph.D. dissertation, Dept. Comput. Sci., Univ. Illinois Chicago, Chicago, IL, USA, 2013.

[21] J. Giles, "Computational social science: Making the links," *Nature*, vol. 488, no. 7412, pp. 448–450, 2012.

[22] D. Easley and J. Kleinberg, *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge, U.K.: Cambridge Univ. Press, 2010.

[23] D. Centola and M. Macy, "Complex contagions and the weakness of long ties," *Amer. J. Sociol.*, vol. 113, no. 3, pp. 702–734, 2007.

[24] K. Saito, M. Kimura, K. Ohara, and H. Motoda, "Efficient estimation of cumulative influence for multiple activation information diffusion model with continuous time delay," in *Proc. 11th Pac. Rim Int. Conf. Artif. Intell.*, Daegu, South Korea, 2010, pp. 244–255.

[25] M. Kimura, K. Saito, and H. Motoda, "Efficient estimation of influence functions for SIS model on social networks," in *Proc. 21st Int. Joint Conf. Artif. Intell.*, Pasadena, CA, USA, 2009, pp. 2046–2051.

[26] K. Saito, M. Kimura, K. Ohara, and H. Motoda, "Learning continuous-time information diffusion model for social behavioral data analysis," in *Proc. 1st Asian Conf. Mach. Learn. Adv. Mach. Learn.*, Nanjing, China, 2009, pp. 322–337.

[27] K. Saito, K. Ohara, Y. Yamagishi, M. Kimura, and H. Motoda, "Learning diffusion probability based on node attributes in social networks," in *Proc. 19th Int. Conf. Found. Intell. Syst.*, Warsaw, Poland, 2011, pp. 153–162.

[28] W. Chen *et al.*, "Influence maximization in social networks when negative opinions may emerge and propagate," in *SIAM Int. Conf. Data Mining*, Mesa, AZ, USA, 2011, pp. 379–390.
[29] C. Budak, D. Agrawal, and A. El Abbadi, "Limiting the spread of misinformation in social networks," in *Proc. 20th Int. Conf. World Wide Web*, Lyon, France, 2011, pp. 665–674.
[30] T. Takaguchi, N. Masuda, and P. Holme, "Bursty communication patterns facilitate spreading in a threshold-based epidemic dynamics," *PLoS ONE*, vol. 8, no. 7, 2013, Art. no. e68629.
[31] A. Clauset, M. E. J. Newman, and C. Moore, "Finding community structure in very large networks," *Phys. Rev. E*, vol. 70, no. 6, 2004, Art. no. 066111.
[32] G. Jeh and J. Widom, "SimRank: A measure of structural-context similarity," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Disc. Data Mining*, Edmonton, AB, Canada, 2002, pp. 538–543.
[33] H. Small, "Co-citation in the scientific literature: A new measure of the relationship between two documents," *J. Amer. Soc. Inf. Sci.*, vol. 24, no. 4, pp. 265–269, 1973.
[34] S. P. Borgatti, "Structural holes: Unpacking Burt's redundancy measures," *Connections*, vol. 20, no. 1, pp. 35–38, 1997.
[35] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proc. Nat. Acad. Sci.*, vol. 105, no. 4, pp. 1118–1123, 2008.
[36] C. Liu, J. Liu, and Z. Jiang, "A multiobjective evolutionary algorithm based on similarity for community detection from signed social networks," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2274–2287, Dec. 2014.
[37] L. Yang, X. Cao, D. Jin, X. Wang, and D. Meng, "A unified semi-supervised community detection framework using latent space graph regularization," *IEEE Trans. Cybern.*, vol. 45, no. 11, pp. 2585–2598, Nov. 2015.
[38] J. Bilmes, "A gentle tutorial on the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models," Int. Comput. Sci. Inst., Univ. Berkeley, Berkeley, CA, USA, Tech. Rep. TR-97-021, 1998.
[39] D. Gruhl, R. Guha, D. Liben-Nowell, and A. Tomkins, "Information diffusion through blogspace," in *Proc. 13th Int. Conf. World Wide Web (WWW)*, New York, NY, USA, 2004, pp. 491–501.
[40] Y. Yang *et al.*, "Rain: Social role-aware information diffusion," in *Proc. 29th AAAI Conf. Artif. Intell.*, Austin, TX, USA, 2015, pp. 367–373.
[41] J. Leskovec. *SNAP: Stanford Network Analysis Platform*. [Online]. Available: http://snap.stanford.edu/snap/index.html, accessed Mar. 10, 2016.
[42] J. Leskovec, L. Backstrom, and J. Kleinberg. *MemeTracker: Download MemeTracker Data*. [Online]. Available: http://memetracker.org/data.html, accessed Mar. 10, 2016.

Qing Bao received the B.Sc. degree in computer science and technology from East China Normal University, Shanghai, China, in 2011. She is currently pursuing the Ph.D. degree in computer science with Hong Kong Baptist University, Hong Kong.

Her current research interests include data mining and social network analysis.

Ms. Bao was a recipient of the Best Student Paper Award in the 2013 IEEE/WIC/ACM International Conference on Web Intelligence.

William K. Cheung received the B.Sc. and M.Phil. degrees in electronic engineering from the Chinese University of Hong Kong, Hong Kong, and the Ph.D. degree in computer science from the Hong Kong University of Science and Technology, Hong Kong, in 1999.

He is an Associate Professor with the Department of Computer Science, Hong Kong Baptist University, Hong Kong. His current research interests include collaborative information filtering, social network analysis and mining, and data mining applications in healthcare.

Dr. Cheung has served as the Co-Chair and a Program Committee Member for a number of international conferences, as well as a Guest Editor of journals on areas, including artificial intelligence, Web intelligence, data mining, Web services, e-commerce technologies, and health informatics. Since 2002, he has been on the Editorial Board of the IEEE Intelligent Informatics Bulletin.

Yu Zhang received the B.Sc. and M.Eng. degrees in computer science and technology from Nanjing University, Nanjing, China, in 2004 and 2007, respectively, and the Ph.D. degree in computer science and engineering from the Hong Kong University of Science and Technology, Hong Kong.

He is a Research Associate with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology. His current research interests include machine learning and data mining, multitask learning, transfer learning, dimensionality reduction, metric learning, and semi-supervised learning.

Dr. Zhang was a recipient of the Best Paper Award in the 26th Conference on Uncertainty in Artificial Intelligence in 2010. He is a Reviewer for various journals and a Program Committee Member for several conferences.

Jiming Liu (F'11) received the M.Eng. and Ph.D. degrees from McGill University, Montreal, QC, Canada, in 1990 and 1994, respectively.

He is the Chair Professor of Computer Science and the Acting Dean of the Faculty of Science, Hong Kong Baptist University, Hong Kong. His current research interests include data analytics, complex systems modeling, and collective intelligence.

Prof. Liu has served as the Founding Editor-in-Chief of the *Web Intelligence Journal* (IOS), and an Associate Editor for *Big Data and Information Analytics* (AIMS), the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, and the IEEE TRANSACTIONS ON CYBERNETICS, AND COMPUTATIONAL INTELLIGENCE (Wiley), among other international journals.