Incremental Learning for Interaction Dynamics with the Influence Model

YongHong Tian, Zheng Mei
Digital Media Lab
Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100080, China
Tel: 86-010-82649529
Email: {yhtian, meiz}@ict.ac.cn

TieJun Huang, Wen Gao
Research Center for Digital Media
Graduate School of Chinese Academy of Sciences
Beijing 100039, China
Tel: 86-010-82649514
Email: {tjhuang, wgao}@jdl.ac.cn

ABSTRACT
Online social networks, which are webs of relationships growing from computer-mediated interactions, have been explored in a wide variety of application domains, such as collaborative information recommendation, collective decision-making, viral marketing plan, etc. In these cases, it is crucial to understand how the networks dynamically affect the users' behaviors. This paper refers to the evolving influence of social networks in the interaction processes as interaction dynamics, and proposes a probabilistic framework for it based on the Influence Model. Moreover, the paper presents a gradient-based algorithm to incrementally learn the model from time-series interaction data, and shows its abilities to characterize chain dependencies through simulation experiments on synthetic data. We also apply the model to mine the dynamic influence networks from the knowledge-sharing sites. The experimental results demonstrate that the proposed model and its learning algorithm can effectively capture the inter-influence relationships between users, and thus can drive the network toward one that has a higher profit potential.

Categories and Subject Descriptors
I.2.6 [**Artificial Intelligence**]: Learning – *induction*; I.5.1 [**Pattern Recognition**]: Models – *statistical*; J.4 [**Computer Applications**]: Social and Behavioral Sciences.

General Terms
Algorithms

Keywords
Online social network, interaction dynamics, the influence model, gradient-based learning algorithm, knowledge-sharing sites

1. INTRODUCTION
With the proliferation of computer networks, more and more interactions take place online. The online social relationship has been one of the most important social relationships between individuals [1]. Growing from computer-mediated interactions, online social networks have been widely applied to promote products in virtual marketing [2, 3], to enhance the collective knowledge and decision-making of small discussion groups [4, 5, 6], to recommend information or guide the search [7], and even to generate new forms of social systems [8]. In these cases, it is crucial to understand how the networks influence the users' behaviors. Studies on this can build a solid foundation for effective applications of online social networks. For example, viral marking [2] can take advantage of networks of influence among customers to inexpen-sively achieve large changes in behaviors. In [2], however, all people in a user's trust web were assumed to have equal influ-ences on his purchasing behaviors. But as a matter of fact, dif-ferent trusted peoples may have different influences on the user's purchasing behaviors, and event the same people at different times may have different influences on his probability of pur-chasing the same products. Therefore, this paper is concerned with how to quantitatively analyze and model the dynamic inter-influence relationships between users in the interaction processes (we call it *interaction dynamics*).

According to the interactivist perspective [9], the internal states of an actor (e.g., person, system, agent, etc.) need to be grounded in interaction histories on the one hand, and have to be related to future interactions on the other hand. Hence, if a state of an actor is characterized on the basis of a set of state properties that do or do not hold at a certain point in time, then an online social interaction process can be viewed as a dynamic process in which the states of all actors have influences on each other over time. In this paper, we therefore model the sequential states of each actor and their corresponding observable behaviors with a hidden Markov model (HMM). Furthermore, we employ the Influence Model [10] to analyze the inter-influencing relationships of all actors on their behaviors. Thus, the interaction dynamics can be represented as an influence model having two inter-influencing processes: the internal state-transition process of each actor and the dynamical inter-influencing process between actors.

To incrementally learn the interaction dynamics, a gradient-based algorithm derived from Coupled HMM (CHMM) [11, 12] is presented in this paper. Simulation experiments show that our algorithm can be more appropriate for incremental learning situations where only limited training data are available at a time than the self-mapping transformation algorithm presented in [12, 13].

In previous work [14], we explored the application of interaction dynamics in the process of information seeking from digital libraries. In this paper, we apply the influence model of interaction dynamics to the domain of knowledge-sharing sites. By extending the network values mining in [2, 3], the experiment was designed to mine the dynamic influence network between the customers from the knowledge-sharing site, Epinions (www. epinions.com). In this experiment, different people in a trust web were differentiated on their network influences, and the influence of the same trusted person who rarely interacted with the user would decay over time. The interaction data of reviewing products were extracted to build the influence model of interaction dynamics. The profit results were compared between the Boolean marketing actions using the webs of trust and that using the influence models. The experimental results shows that our model can effecttively characterize the dynamic influence networks between users, thus has a higher profit potential than the static webs of trust.

The rest of this paper is organized as follows. Section 2 contains a survey of related work. Section 3 introduces the influence model to quantitatively analyze the interaction dynamics. Section 4 presents the incremental gradient-based learning algorithm for the influence model. Experiments on synthetic and real data are described in Section 5. Finally, Section 6 concludes this paper.

2. RELATED WORK

In the last few years there has been a substantial amount of interest in social network dynamics. Some complex statistical models, e.g., discrete- or continuous-time Markov chains [15, 16, 17, 18], were proposed for longitudinal network data. In these models, the network evolution is modeled as the stochastic result of network effects (reciprocity, transitivity, etc.). For estimating and testing these models, Markov chain Monte Carlo (MCMC) methods can be used to develop statistical procedures [15, 16, 19]. In [20], Emily M. Jin *et al.* proposed two simpler models of the growth of social networks based on the principles of clustering, limited degree and friendship decay. Generally, the above models only consider dichotomous relations between actors: the relation (e.g., friendship) from actor *i* to actor *j* either is present, denoted $x_{ij} = 1$, or absent, denoted $x_{ij} = 0$ (the adjacency matrix is called the *sociomatix*). Therefore, these models cannot capture the more microscopic details of social dynamics [20].

To model the influence of social network, weighted influence network models were proposed by Friedkin *et al.* [21, 22], provided that opinions and attitudes of actors in a social system only partially depend on individual characteristics but are also shaped by social influence. The social influence is represented in an influence network, reflecting the dyadic influence of actors on each other. Technically, spatial autocorrelation algorithms are used to capture such processes [23].

The interaction dynamics model needs to combine the ability of Markov chains (or HMM) to model time-series data and the ability of the weighted influence networks to characterize social influence. A natural structure is the Cartesian product HMM [11, 12] or Coupled HMM (CHMM) proposed by Brand, M [10], in which the state of one HMM model at time *t* depends on the states of all models (including itself) at time *t*-1. But there is no parameter within these models to directly characterize the interaction factors. In his PhD thesis [10], C. Asavathiratham developed a special dynamic Bayes net (DBN), the *Influence Model*, which describes the connections between many Markov chains with a simple parametrization in terms of the "*influence*" each chain has on the others. In addition, it shows rich model versatility and analysis methods. Alex Pentland *et al.* [24] used the influence model to quantitatively estimate the interactions between humans in real-life conversational settings. Similarly, this paper employs the influence model to analyze the inter-influencing relationships of all actors on their behaviors.

Meanwhile, many researchers have studied the issue of mining social networks from online interaction data, and explored the applications of these models to different domains. In [4], social networks mined from email logs were used to augment cooperation between users. The ReferralWeb project [7] mined social net-works from a wide variety of publicly-available online information to guide the search for users or documents in response to user queries. In the LambdaMOO MUD, the software agent Cobot [25] utilized the social statistical information gathered from participant interactions to answer queries about these and other usage statistics, and describe the statistical similarities and differences between users. In [2, 3], P. Domingos and M. Richardsons modeled the market as a social network, and used the influence between customers to predict their future purchasing behaviors so as to choose the best viral marketing plan. Inspired by their fruitful work, we also apply the influence model of inter-action dynamics to data mined from Epinions, which was called by them possibly the best-known knowledge-sharing site and an ideal source for experiments on social networks.

3. MODELING INTERACTION DYNAMICS

According to the interactivist perspective [9], the states of an actor need to be grounded in interaction histories. Consider a set of *C* actors that are involving in interactions, and let $s_t^{(c)} (1 \le t \le T)$ be the state variable of the actor *c* at time *t*, which may represent his opinion, attitude or decision. Hence the interaction histories at time *t* are represented by $\{(s_1^{(1)}, \cdots, s_{t-1}^{(1)}), \cdots, (s_1^{(c)}, \cdots, s_{t-1}^{(c)}), \cdots, (s_1^{(C)}, \cdots, s_{t-1}^{(C)})\}$ (which are called *traces* in [9]). In many cases, however, each state does not correspond to an observable event, and can be only estimated from the corresponding observation data. In the context of online interactions, the observation data may be the extracted statistical data, e.g., the interaction frequency and intensity, or the actions that are taken by the users, e.g., the user's rating for the products or purchasing behaviors. Let $x_t^{(c)}$ be the observation variable of the actor *c* at time *t,* and $X = \{(x_1^{(1)}, \cdots, x_T^{(1)}), \cdots, (x_1^{(c)}, \cdots, x_T^{(c)}), \cdots, (x_1^{(C)}, \cdots, x_T^{(C)})\}$ be the time-series of observation variables, therefore the problem is how to estimate the states of the actors from the time-series of observable data, and obtain an insight in the evolution of the states.

A fundamental point for interaction data analysis here is that a state variable $s_t^{(c)}$ at time *t* is assumed to summarize all the information it has before *t*, and the observation $x_t^{(c)}$ at time *t* depends only on the hidden state $s_t^{(c)}$. Moreover, we model the dependency between the sequential states of different actors as a causal Markov random field (MRF). The MRF confers a spatial correlation property between the stochastic structures of multiple sequences, thus can be used an attractive foundation for models of social interactions [27]. Therefore, we have:

$$P(s_t^{(c)} \mid s_1^{(1)},\cdots,s_{t-1}^{(1)},\cdots,s_1^{(c)},\cdots,s_{t-1}^{(c)},\cdots,s_1^{(C)},\cdots,s_{t-1}^{(C)}) = P(s_t^{(c)} \mid s_{t-1}^{(1)},\cdots,s_{t-1}^{(c)},\cdots s_{t-1}^{(C)}) \quad (1)$$

In other words, each state $s_t^{(c)}$ of the actor *c* at time *t* depends on the previous states of all actors (including himself) involving in interactions. According this assumption, the online social interaction can be viewed as a dynamic process in which the states of all actors have influences on each other over time.

Furthermore, let $d_{c'c}$ be the influence strength from the actor *c'* to the actor *c*, which is generally called as the *influence factor*. Then we can model the joint dependency as a linear combination function of all marginal dependencies, i.e.,

$$P(s_t^{(c)} \mid s_{t-1}^{(1)},\cdots,s_{t-1}^{(C)}) = \sum_{c'=1}^{C} d_{c'c} P(s_t^{(c)} \mid s_{t-1}^{(c')}) = d_{cc} P(s_t^{(c)} \mid s_{t-1}^{(c)}) + \sum_{c'=1(c'\neq c)}^{C} d_{c'c} P(s_t^{(c)} \mid s_{t-1}^{(c')}) \quad (2)$$

$P(s_t^{(c)} \mid s_{t-1}^{(c)})$ is the $s_t^{(c)}$'s internal state-transition probability. The external transition probability $P(s_t^{(c)} \mid s_{t-1}^{(c')})$ $(c' \neq c)$ controls how $s_{t-1}^{(c')}$ affects $s_t^{(c)}$, and $d_{c'c}$ determines how much it affects. Note that $\sum_{c'=1}^{C} d_{c'c} = 1$, where d_{cc} measures how self-reliant $s_t^{(c)}$ is. We have also noticed that this formula has the same spirit as the weighted influence network models [21, 22], which assume that opinions and attitudes of actors in a social system only partially depend on individual characteristics but are also shaped by social influence.

More specifically, if the actor *c'* has no influence on the actor *c* (especially, the interaction relationship between *c'* and *c* is absent or unilateral), then $d_{c'c} = 0$. Therefore, in the case of no any interaction with others, $s_t^{(c)}$ evolves with transition probabilities that depend only on his previous state. Namely, the sequential states of each actor and their corresponding observable behaviors can be modeled as a Hidden Markov Model (HMM). The question therefore becomes, how to choose an appropriate model to capture the inter-influence relationships between *C* HMM chains.

As mentioned above, the evolving inter-influencing relationships can be represented as the influence model [10] (See Figure 1). In this figure, one can think of the entire interacting process as a DBN framework having two levels of structure: the network level and the local level. The network level, which is described by a *network graph* $\Gamma(D^T)$ where $D = \{d_{c'c}\}$ is the *influence matrix*, represents the interacting relations between actors. Meanwhile, each actor *c* has a *local HMM chain* $\Gamma(A_c)$ that characterizes the internal state transition in the interacting process. Obviously, the figure 1 felicitously characterizes the dynamics model of online social interactions.

The influence model is specified by the parameters $\lambda = \{\pi, A, B, D\}$, where the initial state probability distribution $\pi = \{\pi_j^{(c)}\}$, the observation probabilities $B = \{b_j^{(c)}(k)\}$, the state-transition matrix $A = \{A_{c'c}\}$, and the influence matrix $D = \{d_{c'c}\}$. All these parameters are subject to stochastic constraints, i.e., $\sum_{j=1}^{N^{(c)}} \pi_j^{(c)} = 1$, $\sum_{k=1}^{M^{(c)}} b_j^{(c)}(k) = 1$, $\sum_{j=1}^{N^{(c)}} a_{ij}^{(c',c)} = 1$ and $\sum_{c'=1}^{C} d_{c'c} = 1$.

Note that if we set $d_{c'c} = 1$ when $d_{c'c} > 0$, and $d_{cc} = 0$ for $1 \le c, c' \le C$, then the influence matrix *D* becomes the sociomatrix, i.e., the standard social network can be viewed as a special case of the influence model of interaction dynamics.

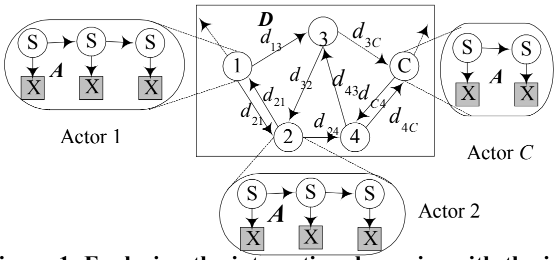


Figure 1. Exploring the interaction dynamics with the influence model (modified from [8]).

4. INCREMENTAL LEARNING FOR THE INFLUENCE MODEL

After the model is specified, the remaining issue is to design its learning algorithm from the time-series interaction data.

In [24], S. Basu *et al.* proposed a learning algorithm for the observed influence model. However, it can be only applied to the cases where observed nodes are strongly interconnected and the hidden states are not. The Distance-Coupled HMM (D-CHMM) proposed by S. Zhong *et al.* [12, 13] provides a tractable alternative for the generalized influence model. As shown in Figure 2, the joint conditional probability in the D-CHMM is modeled as a linear combination of marginal conditional probabilities with the weights represented by coupling coefficients (which is called *distance* between two chains.). Totally, the influence model repre-sented by D-CHMM has $C^2N^2 + C^2$ transition parameters, CN initial probability parameters and CNM observation probability parameters, where *N* is the maximal number of states *per chain* and *M* is the maximal number of observations *per chain*.

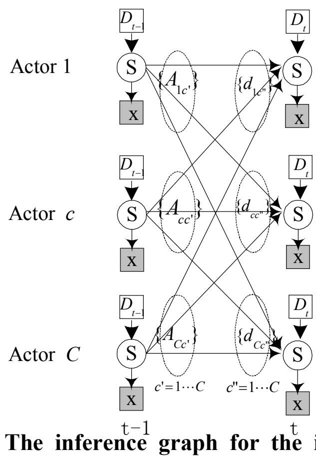


Figure 2. The inference graph for the influence model is represented by the D-CHMM.

Fur-thermore, the literatures [12, 13] presented an iterative optimi-zation procedure for learning the parameters of D-CHMM model based on the self-mapping transformation [12]. This algorithm successfully solved the problem that directly applying EM/GEM algorithm to estimate the D-CHMM parameters is fairly difficult due to the linear combination introduced to model the joint con-ditional dependency. However, similar with the Baum-Welch al-gorithm [26], it also leads to abrupt jumps in parameter space and thus is not suitable for online learning. Therefore, by extending the gradient-based learning algorithm of HMMs to the D-CHMM, we present a gradient-based approach for incrementally learning the influence model from online interaction data.

In the gradient-based approach, any parameter θ is updated ac-cording to the standard formula:

$$\theta^{new} = \theta^{old} - \eta \left[\frac{\partial J}{\partial \theta} \right]_{\theta = \theta^{old}} \tag{3}$$

where J is a quantity to be minimized, and η is the learning rate. We define in this case,

$$J = -\log(P(X \mid \lambda)) \tag{4}$$

where the likelihood function $P(X \mid \lambda)$ can be calculated by the *extended forward-backward procedure* presented in [12, 13]:

a. Initialization: $\alpha_1^{(c)}(j) = \pi_j^{(c)} \cdot b_j^{(c)}(x_1^{(c)})$ (5)

b. Induction:

$$\alpha_t^{(c)}(j) = b_j^{(c)}(x_t^{(c)}) \sum_{c'=1}^{C} d_{c'c} \sum_{i=1}^{N^{(c')}} \alpha_{t-1}^{(c')}(i) \cdot a_{ij}^{(c',c)} \tag{6}$$

c. Termination: $P(X \mid \lambda) = \prod_{c=1}^{C} P^{(c)} = \prod_{c=1}^{C} \left(\sum_j \alpha_T^{(c)}(j) \right)$ (7)

where $\alpha_t^{(c)}(j)$ is the extended forward variable. Therefore,

$$J = -\sum_{c=1}^{C} \log P^{(c)} = -\sum_{c=1}^{C} \log \left(\sum_s \alpha_T^{(c)}(s) \right) \tag{8}$$

Then the problem is to find the derivative $\frac{\partial J}{\partial \theta}$ for any parameter θ of the model. For the influence factor d_{ji}, we have:

$$\frac{\partial J}{\partial d_{ji}} = -\frac{1}{P} \cdot \frac{\partial P}{\partial d_{ji}} = -\sum_c \left(\frac{1}{P^{(c)}} \sum_{s=1}^{N^{(c)}} \frac{\partial \alpha_T^{(c)}(s)}{\partial d_{ji}} \right) \tag{9}$$

To overcome the absorption problem of 0 probabilities, Baldi and Chauvin [26] used a normalized-exponential representation of the parameters for smoothly online learning. The normalized-expo-nential representation of d_{ij} with the constraints $\sum_{j=1}^{C} d_{ji} = 1$ and $d_{ji} \geq 0$ is fixed at each iteration as:

$$d_{ji} = \frac{e^{\mu \omega_{ji}}}{\sum_k e^{\mu \omega_{ki}}} \tag{10}$$

where μ is a temperature parameter which can be absorbed in the learning rate. Then it is easy to verify the following deri-vatives:

$$\frac{\partial d_{ki}}{\partial \omega_{ji}} = \mu d_{ki} (\delta_{j,k} - d_{ji}) \tag{11}$$

and

$$\frac{\partial J}{\partial \omega_{ji}} = \sum_k \left(\frac{\partial J}{\partial d_{ki}} \cdot \frac{\partial d_{ki}}{\partial \omega_{ji}} \right)$$

$$= -\mu \sum_c \frac{1}{P^{(c)}} \left[d_{ji} \sum_{s=1}^{N^{(c)}} \frac{\partial \alpha_T^{(c)}(s)}{\partial d_{ji}} - d_{ji} \sum_k d_{ki} \sum_{s=1}^{N^{(c)}} \frac{\partial \alpha_T^{(c)}(s)}{\partial d_{ki}} \right]$$

$$= -\mu \sum_c \frac{1}{P^{(c)}} \left[t^{(c)}(d_{ji}) - d_{ji} t^{(c)}(d_i) \right], \tag{12}$$

where $\delta_{x,y} = \begin{cases} 1, x = y \\ 0, x \neq y \end{cases}$, $t^{(c)}(d_{ki}) = d_{ki} \sum_{s=1}^{N^{(c)}} \frac{\partial \alpha_T^{(c)}(s)}{\partial d_{ki}}$, $t^{(c)}(d_i) = \sum_k t^{(c)}(d_{ki})$. As discussed in [12], the first derivative of $\partial \alpha_t^{(c)}(s) / \partial d_{ji}$ can be calculated using back-propagation through time, e.g.

$$\frac{\partial \alpha_t^{(c)}(s)}{\partial d_{ji}} = \begin{cases} 0, t = 1 \\ \delta_{c,i} \sum_k a_{ks}^{(j,i)} b_s^{(i)}(x_t^{(i)}) \alpha_{t-1}^{(j)}(k) + \sum_{c'} \sum_k z_{kst}^{(c',c)} \frac{\partial \alpha_{t-1}^{(c')}(k)}{\partial d_{ji}}, t > 1 \end{cases} \tag{12}$$

where $z_{ijt}^{(c',c)} = d_{c'c} \cdot a_{ij}^{(c',c)} \cdot b_j^{(c)}(x_t^{(c)})$.

Finally, we get the d_{ji}'s update equation as follows:

$$\omega_{ji}^{new} = \omega_{ji}^{old} + \eta \sum_c \frac{1}{P^{(c)}} \left[t^{(c)}(d_{ji}) - d_{ji} t^{(c)}(d_i) \right] \tag{13}$$

where μ has be merged in the learning rate η ($0 \leq \eta \leq 1$, typically $\eta = 0.9$, See Fig. 3). Similar arguments can be made for π, A, B.

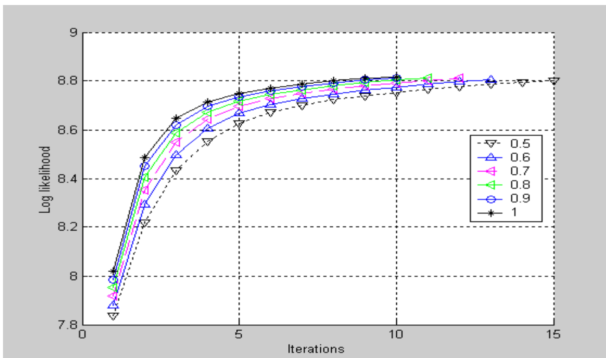


Figure 3. The Graph shows the likelihood curves in learning process when the learning rate varies from 0.5 to 1.0. To trade-off the speed of convergence and stability of learning process, the learning rate is set to be 0.9 in our experiments.

5. EXPERIMENTS AND RESULTS

We tested the influence model in several experiments on synthetic data to gauge the performance of its gradient-based learning algo-rithm, then on Epinions data to gather empirical results.

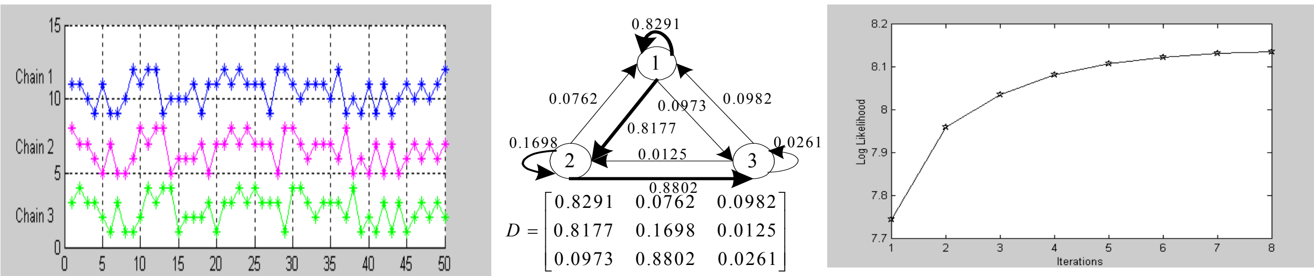
5.1 Experiments on Synthetic Data

To test the influence model on synthetic data, a 3-chains influence model with 3 hidden states and 4 observation states *per chain* was used to generate training sequences. In order that the training se-quences can be sampled in the queuing model (i.e., chain 1 is evolving randomly, and chain 2 meticulously follows chain 1, and in turn chain 3 follows chain 2), the parameters were generated as follows: The priors π were initialized at random; The influence matrix D was set to be $\begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$; And all chains used the same transition matrix and observation matrix, which were also gene-rated randomly.

These training sequences were then used to train another randomly initialized influence model. The gradient-based algorithm was utilized to estimate the model parameters. Figure 4 shows a group of training observation sequences, the resulting influence network and its learning log likelihood curve. It can be seen that the learned influence factors can exactly capture the following behavior implied in the observation sequences. We notice that this training process converged fairly rapidly by only 8 iterations, as depicted by the log likelihood curve in the same figure.

We also compared the learning performance and the speed of convergence between our approach and the self-mapping transformation algorithm described in [12, 13]. For learning performance, we look at the log likelihood of fitting training data to the trained model. The higher the likelihood, the better local maximum we think the training has converged to. The speed of convergence is measured by the iteration times and processing time. Figure 5 shows the experimental results. Not surprisingly, the gradient-based approach demonstrated better performance and rapider convergence speed than the self-mapping transformation algorithm on small sizes of training sets. With more training data available, the self-mapping transformation algorithm gradually showed its advantages in the speed of convergence. But in many cases, it's not easy to obtain sufficient training data for the model at a time so that the data need to be incrementally collected, thus the gradient-based algorithm is more suitable for the parameter training tasks in these cases.

The learning rate and the size of data window are two most important parameters in the gradient-based learning algorithm. The former has been discussed in the previous section. And for the latter, figure 6 depicts the learning performance and the speed of convergence when the size of data window varies from 2 to 10 in the training processes. It can be seen that despite the learning performance has some improvement with enlarging the size of data window, the convergence speed and process time go up synchronously due to the increase of computational complexity of the forward-backward procedure in each data window. However, after the size is larger than 8, they will decline along with the decrease of total iteration times. A reasonable explanation is that



(a) (b) (c)

Figure 4. Test the influence model on the synthetic training sequences: (a) the training observation sequences; (b) the learned influence network and its corresponding influence matrix; (c) the log likelihood curve of the training process.

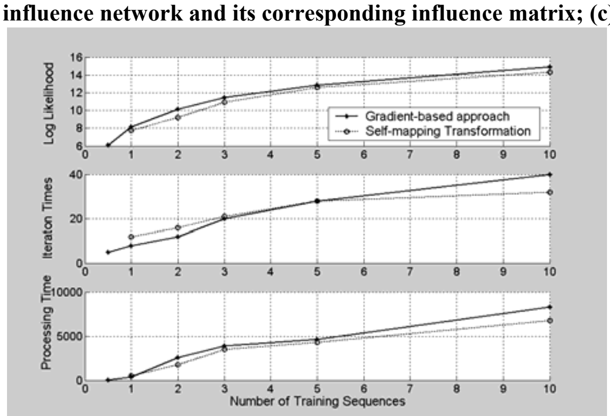


Figure 5. Compare the likelihood, the iteration times and processing time between the gradient-based approach and self-mapping transformation algorithm on small sizes of training sets.

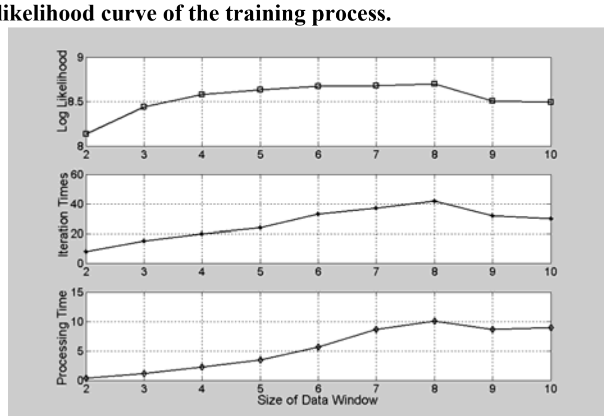


Figure 6.Compare the training likelihood, the iteration times and processing time between different sizes of data windows.

with the enlargement of the window size, more statistical features captured from the observation data in each window can accelerate the convergence. Considering the trade-off between learning performance and the speed of convergence, the size of data window is set to be 2 in the other experiments.

Summarily, the simulation experiments show the abilities of the influence model to characterize chain dependencies. Moreover, the experiments also demonstrate that the gradient-based algorithm can be more appropriate for incremental learning situations where only limited training data are available at a time than the self-mapping transformation algorithm.

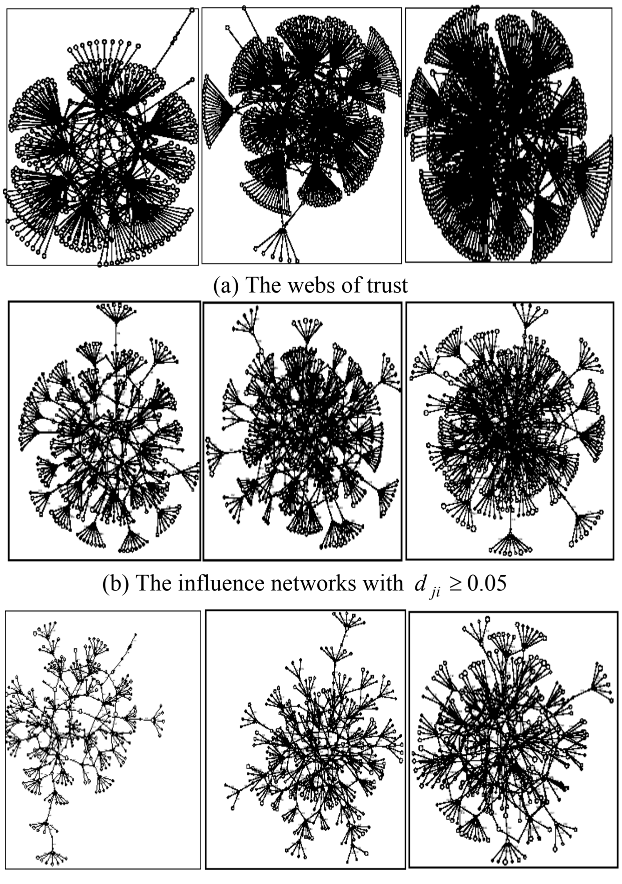
5.2 Experiments on Real Data

Arguably, a decade ago it would have been difficult to make practical use of the interaction dynamics model for the lack of data to estimate the influence probabilities [3]. In previous work [14], we designed a small-scale experiment to explore the interaction dynamics in the process of information seeking from digital libraries. In that context, a well-tuned influence network of online interactions can enhance the group's collective knowledge and sharpen its ability to act on what people know in time to be effective. Similar application settings can be found in online education or collective decision-making within an organization.

In [2, 3], P. Domingos and M. Richardsons mined social networks from collaborative filtering databases and knowledge-sharing sites, and used the influence between customers to predict their future purchasing behaviors so as to guide viral marketing more targetedly. Inspired by their fruitful work, we also apply the influence model of interaction dynamics to data mined from Epinions, which was called by them possibly the best known knowledge-sharing sit and an ideal source for experiments on social networks. Different with their work, however, this paper concentrates on how to more accurately characterize the influence between the users' purchasing behaviors, and how to trace the evolution of the influence networks between users from interaction data. Of course, the learned dynamic influence models will be used to calculate the network effects on the *expected lift in profits* (ELP) of viral marketing to demonstrate their advantages over the webs of trust. And we do not attempt to capture more detailed topics, e.g., continuous marketing scenarios, acquiring new network knowledge, and so on.

On Epinions, members submit product reviews, and give a rating of zero to five stars to any product. Epinions users read these reviews, rate them according to how helpful or accurate they are, and even directly write comments on them. If a user is interested in a product or has a high potential probability of purchasing the product, he can subscribe to review alerts for the product. Meanwhile, users may directly rate other users and list reviewers that they trust. In [2], this kind of the trust webs was directly used to model the network effect on the user's purchasing behaviors. In Epinions, all interaction data and user data are recoded in archives according to their release date (Date added information is not available for members trusted/blocked prior to Jan 11, 2001). For simplicity, we obtained our experimental data by crawling through the pages of the most popular authors in the product category "Computer Hardware". The obtained data were divided into seven subsets according to their release date, i.e., $X_t (t = 0, \cdots, 6)$, where X_0 was the set of the data prior to Jan 11, 2001, and the other data between Jan 11, 2001 and Jul 1, 2003 were evenly divided into six subsets (i.e., a subset *per* half a year).

We considered the following principles for conducting our experiment. In [2], all trusted people of a user were always assumed to have equal influences on him. However, different trusted people may have different influences on the user's purchasing behaviors, and even the same people at different times may have different influences on his probability of purchasing the same products. Therefore, we emphasized here that different people in a trust web were differentiated on their network influences, and the influence of the same trusted person who rarely interacted with the user would decay over time. The latter assumption is based on the obvious mechanism in real-life acquaintanceship [20]: Even after two people become acquainted, they still need to meet regularly in order to maintain that acquaintance. In cases of networks, the trust friendship even decays at a higher rate, especially for individuals who have not been acquainted with each other in real-life. In Epinions, a kind of good informative data for the activity of the relationship among two users is the times that they deliberated or compared reviews about the products in recent past. For example, it's reasonable to assign a high influence weight from the user *j* to *i* if the user *i* often gives comments or direct ratings on *j*'s reviews that are related to some products the user *i* would



(a) The webs of trust

(b) The influence networks with $d_{ji} \geq 0.05$

(c) The learned influence networks with $d_{ji} \geq 0.1$

Figure 7. Compare the leaned influence networks with the webs of trust for the 100 most popular reviewers in the *Computer Hardware* category from the year of 2000 to 2002.

like (In our collected data, the average number of comments *per* review is 2.9).

We also placed a limit on the number of trusted people who could influence a user's purchasing behaviors. In real-life, there is a recurring cost in terms of time and effort to maintaining a friendship [20]. Though the cost will be decrease dramatically in cases of networks, and the distribution of trust relationships in the web of trust in Epinions is Zipfian [2], we can not imagine that when a user is inclined to purchase a product, his decision is affected by as many as 506 trusted users (which is the maximal size of trust web in our gathered data). Therefore, this experiment used the evolving influence networks rather than the webs of trust to model the inter-influence relationships between customers, and set 50 as the maximal in-degree of each node (the in-degree of a node indicates how many nodes affect it). Though might be initialized by the webs of trust, the influence network has three distinct features, i.e., limited in-degree of nodes, weighted influence, and decay of influence. Figure 7 depicts the comparison between the learned influence networks with two kinds of influence thresholds and the webs of trust for the 100 most popular reviewers in the Computer Hardware category from the year of 2000 to 2002. Note that for the limitation of drawing space, the figure 7(a) only depicted the interconnected network generated by partial trust webs for top 20 users, each of which only contained the newly-added 50 trusted people in each year, and the figure 7 (b) and (c) ignored some small clusters that were disconnected with the main components of the influence networks.

To apply the influence model to Epinions, we needed to estimate some parameters, including the priors, the transition probabilities, the observation probabilities and the influence factors. The meaning of the variables in the Epinions is similar to that described in [2] and [3]: $s_t^{(c)}$ $(t=0,\cdots,6)$ is whether the user c buys the product at time t being considered. $x_t^{(c)}$ contains the observation data related to the user c at time t, such as his rating given to the product, $r_t^{(c)}$, the interaction statistics data, $is_t^{(c)}$, and the product attributes Y_t. We assumed that given $s_t^{(c)}$, the three variables $r_t^{(c)}$, $is_t^{(c)}$ and Y_t were independent, i.e.,

$$P(x_t^{(c)} \mid s_t^{(c)}) = P(r_t^{(c)} \mid s_t^{(c)}) P(is_t^{(c)} \mid s_t^{(c)}) P(Y_t \mid s_t^{(c)}) \quad (14)$$

For simplicity, we also assumed that all users in the same product category had the similar behavior manners so that we could need to estimate only one state-transition matrix and one observation matrix for them. In our experiments, all parameters except the influence factors could be initialized by the similar methods proposed in [2, 3]. For instance, the prior $P(s_0^{(c)})$ could be estimated simply as the fraction of products rated by the user c; $P(Y_t \mid s_t^{(c)})$ could be obtained by counting the number of occurrences of each value of Y_0 with each value of $s_0^{(c)}$ at time t_0; $P(r_t^{(c)} \mid s_t^{(c)})$ could be estimated by the piecewise-linear model; and transition probability $P(s_t^{(c)} \mid s_{t-1}^{(c)})$ might be initialized at random. For the influence factors, we initialized them in the following two cases:

- The self-reliance d_{cc} was initialized to 0.5, as in [2].
- The influence factor $d_{c'c}$ from the user c' to c was estimated by counting the frequency of the comments or ratings that the user c gave on the reviews of the user c'. That is, if $f_0^{(c',c)}$ denoted the times of the comments or ratings that the user c gave on the reviews of the user c' at time t_0, and $f_0^{(c)} = \sum_{\forall c''} f_0^{(c'',c)}$, then $d_{c'c}$ was initialized as follows:

$$d_{c'c} = \frac{f_0^{(c',c)}}{f_0^{(c)}} = \frac{f_0^{(c',c)}}{\sum_{\forall c''} f_0^{(c'',c)}} \quad (15)$$

Note that $d_{c'c} = 0$ for any c', if c was an inactive user who rarely wrote any own review or comment. So for any inactive user c, we simply set $d_{c'c} = 0.5 * (1/|N^{(c)}|)$ if $c' \in N^{(c)}$, and $d_{c'c} = 0$ if $c' \notin N^{(c)}$, where $N^{(c)}$ was the user set in the c's trust web.

Once all the parameters are initialized, the gradient-based learning algorithm was performed to iteratively re-estimating probabilities until they all converged. The figure 7 (b) and (c) illustrate the influence networks learned in the incremental leaning process. It should be noted that the dependencies between nodes in the learned influence networks are much less than those in the webs of trust. This is because our mode exactly captures the dynamic changes of the influences between individuals from the interaction data. And with more interaction information available as time passed, more influence relationships were captured so that the learned influence networks appeared a little more complex. In this process, some new influence relationships would come forth, and some old ones would strengthen further while some would decay.

Table 1: The Boolean viral marketing profit results for the two network effect models

	$\alpha=2, r_0=1, r_1=1$		
	c=0.1	c=0.01	c=0.001
Using the web of trust	6.31	7.79	10.27
Using the influence model	7.28	8.73	10.85
Lift in profit	0.97 (15.4%)	0.94 (12.1%)	0.58 (5.6%)

We also repeated the Boolean marketing experiments described in [2], respectively using the webs of trust and the influence models to calculate the network effects. The data X_t $(t=0,\cdots,5)$ were used as the training set, and the data X_6 was used as the test set. All the parameters of the Boolean viral marketing model were set similarly with those in [2]. Table 1 shows the viral marketing pro-fit results for the two network effect models. It can be seen that viral marketing using the influence models had obtained some increase in profit over the case using the webs of trust. However, with decreasing c, the rate of lift in profit declined gradually. A possible reason is that a lower cost of marketing will increase the profit obtained by marketing to those with small influences on the others.

The experimental results shows that our model and its learning algorithm can effectively capture the changing influence network structure between customers during online social interactions, thus drives the network toward one which has a higher profit potential [2]. However, the lift would be higher if more informative interaction data are available, especially for those inactive users who rarely wrote any own review or comment. We will explore this issue in the future work.

6. CONCLUSION

In this paper, we aim at developing a probabilistic framework to model the interaction dynamics for online social interactions and investigating its effective learning algorithms. We employed the influence model to analyze the dynamic inter-influencing relationships between actors, and proposed a practical incremental learning algorithm for our model. Experiments on simulated and real data verified that the proposed model could effectively characterize the dynamic inter-influence between users during on-line interaction processes.

We believe that our work is an important step towards further understanding the roles of interactions and social networks, though it is impossible to capture all of the subtleties with a simple model. In future, we will further study how to exploit this model to analyze the influence of a substantial amount of information access behaviors on the document semantic space of the multimedia database from the viewpoint of the interaction between the user and the system.

7. ACKNOWLEDGEMENTS

This work was supported by China-America Digital Academic Library Project, and also by "Knowledge Innovation Initiative" of Chinese Academy of Sciences under Grant No. Kgcxz-103.

8. REFERENCES

[1] B. Wellman. Computer Networks As Social Networks. *Science* Vol 293(14): 2031-2034, SEP 2001.

[2] M. Richardson and P. Domingos. Mining Knowledge-Sharing Sites for Viral Marketing. In *Proceedings of the Eighth International Conference on Knowledge Discovery and Data Mining (SIGKDD'02)*, Edmonton, Alberta, Canada: 61-70, 2002. ACM Press.

[3] P. Domingos and M. Richardson. Mining the Network Value of Customers. In *Proceedings of the Seventh International Conference on Knowledge Discovery and Data Mining (SIGKDD'01)*, San Francisco, CA: 57-66, 2001. ACM Press.

[4] L. Sproull and S. Kiesler. Connections. New Ways of Working in the Networked Organization. Cambridge, MA, United States: The MIT Press, 1991.

[5] S. D. Farnham, H. Chesley, D. McGhee, and R. Kawal. Structured On-line Interactions: Improving the Decision-making of Small Discussion Groups. In *Proceedings of CSCW 2000*, Philadelphia, Pennsylvania, United States, 2000: 299–308.

[6] H. Ogata, Y. Tano, N. Furugori and Q. Jin. Computer Supported Social Networking for Augmenting Cooperation. *Computer Supported Cooperative Work*, Vol. 10:189-209, 2000.

[7] K. Kautz, B. Selman, and M. Shah. ReferralWeb: Combining social networks and collaborative filtering. *Communications of the ACM*, 40(3): 63-66,1997.

[8] B. Becker and G. Mark. Constructing social systems through computer-mediated communication. In: *Virtual Reality Society Journal*, Churchill, E. Snowdon, D. (Hrsg.), Springer Verlag, London: 60–73, 1999.

[9] C. M. Jonker and J. Treur. A Temporal-Interactivist Perspective on the Dynamics of Mental States. *Cognitive Systems Research Journal*. To appear, 2003.

[10] C. Asavathiratham. The Influence Model: A Tractable Representation for the Dynamics of Networked Markov Chains. [PH.D thesis] Dept. of EECS, Massachusetts Institute of Technology. Cambridge, United States, 2000.

[11] M. Brand, N. Oliver and A. Pentland. Coupled Hidden Markov Models for Complex Action Recognition. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, Puerto Rico: 994–999, 1997.

[12] S. Zhong and J. Ghosh. A New Formulation of Coupled Hidden Markov Models Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, United States, Tech. Report, 2001

[13] S. Zhong and J. Ghosh. Distance-coupled Hidden Markov Models. Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, United States, Tech. Report, 2001

[14] Y. H. Tian, T. J. Huang, W. Gao. The Influence Model of Online Social Interactions and its Learning Algorithms. *Chinese Journal of Com-puters*, Vol. 28, No. 7: 848-858, Jul 2003.

[15] T.A.B. Snijders, and M.A.J. van Duijn. Simulation for statistical inference in dynamic network models. In: Conte, R., Hegselmann, R., and Terna, P. (Eds.), *Simulating Social Phenomena.* Lecture Notes in Economics and Mathematical Systems, 456. Berlin: Springer: 493-512, 1997,

[16] T.A.B. Snijders. The Statistical Evaluation of Social Network Dynamics. Sociological Methodology, 2001

[17] P. Holland and S. Leinhardt. A dynamic model for social networks. *Journal of Mathematical Sociology*, Vol. 5: 5-20, 1977.

[18] S. Wasserman. Analyzing social networks as stochastic processes. *Journal of the American Statistical Association,* Vol. 75: 280-294, 1980.

[19] T.A.B. Snijders. Stochastic actor-oriented models for network change. *Journal of Mathematical Sociology*, Vol. 21: 149-172, 1996.

[20] M. J. Emily, G. Michelle, and M. E. J. Newman, The structure of growing social networks, *Physics Review*. E 64, 046132, 2001.

[21] N.E. Friedkin and E.C. Johnsen. Social influence and opinions, J*ournal of Mathematical Sociology*, Vol. 15: 193-205, 1990.

[22] P.V. Marsden and N.E. Friedkin. Network studies of social influence. *Sociological Methods & Research*, Vol. 22: 127-151, 1993.

[23] R.Th.A.J. Leenders. Modeling social influence through network auto-correlation: constructing the weight matrix. *Social Networks,* Vol. 24: 21-48, 2002.

[24] S. Basu, T. Choudhury, B. Clarkson, and Pentland, A. Towards Measuring Human Interactions in Conversational Settings. In *Proceedings of the IEEE Int'l Workshop on Cues in Communication (CUES 2001) at CVPR 2001*, Kauai, Hawaii, United States, 2001.

[25] C. L. Isbell, Jr., M. Kearns, S. Singh, and P. Stone, Cobot in LamdaMOO: A Social Statistics Agent. In *Proceedings of the 17th National Conference on Artificial Intelligence*, Austin, TX: 36-41, 2001

[26] P. Baldi and V. Chauvin. Smooth On-line Algorithms for Hidden Markov Models. *Neural Computation,* Vol. 6:307–318, 1994.

[27] Y. Ioannides. Topologies of Social Interactions, Discussion Papers Series, Departement of Economics, Tufts University. 2002.