Modeling Dynamical Influence in Human Interaction

Wei Pan, Wen Dong, Manuel Cebrian, Taemie Kim, Alex (Sandy) Pentland

Media Laboratory, MIT

20 Ames Street, Cambridge, MA 01239, USA

{panwei, wdong, cebrian, taemie, pentland}@media.mit.edu

March 31, 2011

Abstract

How can we model influence between individuals in a social system? How can we use influence to model and predict observations from social systems? In this article, we explain the recent advances of the influence model, a Bayesian network approach for modeling social influence from observations of individuals. We review the development of the influence model in the literature. We also introduce the generalization of the influence model, the dynamical influence model, and demonstrate three examples on how to use these models.

1 Introduction

For decades, social scientists and psychologists have been interested in analyzing and understanding *who influences whom* in a social system, such as a group discussion process[1, 2, 3, 4]. Influence is also particularly interesting in the context of leadership and group dynamics, where the influence between one another has been recognized as a significant factor of group performance[5].

However, from a signal processing and modeling point of view, it remains a difficult question to formally define *influence* in a mathematical way. It is also common that the influence between individuals is not directly observable, and only individual-level behavioral signals are generally available. For instance, many modern sensing systems such as the sociometric badges [6] and cell phones [7] now provide valuable social behavioral signals from each individuals. Therefore, the challenges are to not only define *influence*, but also infer *influence* from individual observations and individual signals.

A line of research works, known as the influence model, is focused on modeling social influence mathematically. The influence model addresses two fundamental challenges:

- The influence model mathematically defines "influence" and how influence changes, and the "influence" learned by the influence model is tightly connected with the sociological meaning of influence.
- The influence model enables researchers to infers interactions and influence dynamics by only using time series signals from individual observations.

This article is organized in the following way: We give an overview for the formulation of the influence model in Section 2, and we discuss previous works in the review section (Section 3). We continue to introduce the dynamical influence model, a generalization for the influence model for changing influence and network structure described in Section 4. We will discuss the inference algorithm for these model in Section 5. Finally, we explain a few examples in detail on artificial and real data for the dynamical influence model in Section 6.

2 Overview for the Influence Model

2.1 Entities in a Social System

We describe the formulation of the influence model here, followed by a review on its history in Section 3. The model starts with a system of C entities. Each entity can be a person in a group discussion, or a geographical district in modeling flu epidemics. We assume that each entity c is associated with a finite set of possible states $1, \ldots, S$. At different time t, each entity c is in one of the states, denoted by $h_t^{(c)} \in \{1, \ldots, S\}$. It is not necessary that each entity is associated with the same set of possible state. Some entities can have more or less states. However, to simplify our description, we assume that each entity's latent state space is the same without loss of generality. The state of each entity is not directly observable. However, as in the Hidden Markov Model (HMM), each entity emits a signal $O_t^{(c)}$ at time stamp t based on the current latent state $h_t^{(c)}$, following a conditional emission probability $\text{Prob}(O_t^{(c)}|h_t^{(c)})$. The emission probability can either be multinomial or Gaussian for discrete and continuous cases respectively, exactly as in HMM literature [8].

2.2 Influence between Entities

The influence model is composed of entities interacting and influencing each other. "Influence" is defined as the conditional dependence between each entity's current state $h_t^{(c)}$ at time t and the previous states of all entities $h_{t-1}^{(1)}, \ldots, h_{t-1}^{(C)}$ at time $t-1$. Therefore, intuitively, $h_t^{(c)}$ is *influenced* by all other entities. We now discuss the conditional probability:

$$\text{Prob}(h_t^{(c')}|h_{t-1}^{(1)}, \ldots, h_{t-1}^{(C)}). \tag{1}$$

Once we have $\text{Prob}(h_t^{(c')}|h_{t-1}^{(1)}, \ldots, h_{t-1}^{(C)})$, we naturally achieve a generative stochastic process.

As in the coupled Markov Model [9], we can take a general combinatorial approach Eq. 1, and convert this model to a equivalent Hidden Markov Model (HMM), in which each different latent state combination of $(h_{t-1}^{(1)}, \ldots, h_{t-1}^{(C)})$ is represented by a unique state. Therefore, for a system with C interacting entities, the equivalent HMM will have a latent state space of size S^C, exponential to the number of entities in the system, which seems to be unacceptable in real applications.

The influence model approach, on the other hand, uses a much simpler mixture approach with far fewer parameters. Entities $1, \ldots, C$ influence the state of c' in the following way:

$$\text{Prob}(h_t^{(c')}|h_{t-1}^{(1)}, \ldots, h_{t-1}^{(C)}) = \sum_{c \in \{1,\ldots,C\}} \underbrace{\mathbf{R}_{c',c}}_{\text{tie strength}} \times \underbrace{\text{Prob}(h_t^{(c')}|h_{t-1}^{(c)})}_{\text{cond. probability}}, \tag{2}$$

where $\mathbf{R}$ is a $C \times C$ matrix. ($\mathbf{R}_{c_1,c_2}$ represents the element at the c_1-th row and the c_2-th column of the matrix $\mathbf{R}$) $\mathbf{R}$ is row stochastic, i.e., each row of this matrix sums up to one. $\text{Prob}(h_t^{(c')}|h_{t-1}^{(c)})$ is modeled using a $S \times S$ row stochastic matrix $\mathbf{M}^{c,c'}$, so that $\text{Prob}(h_t^{(c')}|h_{t-1}^{(c)}) = \mathbf{M}^{c,c'}_{h_{t-1}^{(c)}, h_t^{(c')}}$. This matrix $\mathbf{M}^{c,c'}$ is also known as the transition matrix in HMM literature [8]. Generally, for each entity c, there are C different transition matrices in the influence model to model the influence dynamics between c and $c', c' = 1, \ldots, C$. However, it can be simplified by replacing the C different matrices with only two $S \times S$ matrices $\mathbf{E}^c$ and $\mathbf{F}^c$: $\mathbf{E}^c = \mathbf{M}^{c,c}$, which captures self state transition; We also assume that the influence of entity c over other nodes are the same, so they can be replaced with on single matrix $\mathbf{F}$, i.e. $\mathbf{M}^{c,c'} = \mathbf{F}^c, \forall c' \neq c$.

Eq. 2 can be viewed as follows: all entities' states at time $t-1$ will influence the state of entity c' at time t. However, the strength of influence is different for different entities: the strength of c over c' is captured by $\mathbf{R}_{c',c}$. As a result, the state distribution for entity c' at time t is a combination of influence from all other entities weighted by their strength over c'. Because $\mathbf{R}$ captures influence strength between any two entities, we refer to $\mathbf{R}$ as *Influence Matrix*.

2.3 Inference

Therefore, the influence model is a generative model defined by parameters $\mathbf{R}, \mathbf{E}^{1:C}, \mathbf{F}^{1:C}$ and the emission probabilities $\text{Prob}(O_t^{(c)}|h_t^{(c)}), \forall c$. As most generative machine learning models, these parameters are not set by users, but the are automatically learned from observations $O_{1:T}^1, \ldots, O_{1:T}^C$. The inference algorithms for learning these parameters will be discussed in Section 4.

2.4 Remarks for the Influence Model

1. The number of parameters in our model grows quadratically with respect to the number of entities C and the latent space size S. This largely relieves the requirement for large training sets and reduces the chances of model overfitting. As a result, the influence model is more scalable for larger social systems, and is resistant to overfitting when training data is limited compared with other approaches [10].

2. This approach captures the tie strength between entities using a $C \times C$ matrix $\mathbf{R}$. $\mathbf{R}$ can be naturally treated as the adjacency matrix for a directed weighted graph in graph theory. The influence strength between two nodes learned by our model can be then treated as tie weights in social networks. This key contribution connects the conditional probabilistic dependence to a weighted network topology. In fact, in previous works, the most common usage for the influence model is to use $\mathbf{R}$ to understand social structure [11, 12] .

3 Review

The influence model has been developed and applied to model different aspects of social systems. Many works on the influence model started with the development of sociometric badges as shown in Fig. 1, a personal device collecting individual behavioral data such as audio, movements, etc. The sociometric badges provide rich measurements for individuals, and the question of modeling group interaction and influence from individual observation signals naturally rose.

Figure 1: Different versions of the sociometric badge is shown in the left and in the middle. The sociometric badge is a wearable sensing device for collecting individual behavioral data. On the right is a group brainstorming session, and all participants were wearing the sociometric badges.

The influence model is introduced in [13], in which the authors firstly developed this idea around modeling influence, an inference scheme based on optimization, and a successful application of studying the audio recording from a group discussion session with five individuals. The reseachers used audio features as observations $O_t^{(c)}$ for the influence model, and modeled the latent state space to be either “speaking” or “non-speaking”. The influence model bridges between the noisy signal processing measured directly from each individual to the underlying inter-personal influence and interaction on turn taking.

One key question for the influence model is that if the influence matrix $\mathbf{R}$ represents well the concept of influence in human interactions. In other words, if the definition of influence in this article has practical and sociological meanings. Using the conversation data from the wearable sociometric badges on 23 individuals, it was discovered that the influence strength between individuals learned by influence model correlates extremely well with individual centrality in their social networks(with $R = 0.92, p < 0.0004$)[12]. This evidence suggests that the influence matrix defined as the weights in the conditional dependence on states of other entities is an important measure for the influence of the individual in real social interactions.

The influence model is then applied to many different human interaction problems. For instance, researchers have used the influence model in understanding the functional role (follower, orienteer, giver, seeker, etc) of each individual in the mission survival group discussion dataset[14]. Researchers discovered that using the influence matrix they were able to achieve better classification accuracy compared with other approaches. By using the Reality Mining[7] cellphone sensor data from 80 MIT members as observations, and constraining the latent space of each individual to be binary “work” and “home”, researchers found that the influence matrix learned from this data matches well with the organizational relationship between individuals[10]. This is intuitive as students’ schedules are likely to be influenced by close colleagues’ working schedule. Also, related works[11] often use the influence model as a measurement tool for social systems. The influence model has been extended to model other systems such as the traffic system[15] and the flu epidemics among US states[16].

The influence model is extended to be suitable for dynamical situation, in which the influence matrix itself changes[16]. This new approach, the Dynamical Influence Model, is a generalization of the inference model, and is discussed in the following section.

3.1 Other Approaches

For other related approach, the Bayesian network is a tool often used in understanding and processing social interaction time series data. Earlier projects have used coupled HMM [9], and more recent projects have used dynamic system trees [13] and interacting Markov chains [17]. The key contribution and difference of the influence model is that we use the influence matrix $\mathbf{R}$ to connect the social network to state dependence.

Other relevant general multi-dimensional time series approaches such as LDS [18] and the prototype model [19] are not able to recognize the network structure and weights on edges between nodes in social systems.

Defining influence as the state dependence for an entity on states of other nodes is an idea that has been extensively explored by the statistical physics society as well. Castellano et al[20] refer to these statistical physics models as “opinion dynamics”.

4 Introduction to Dynamical Influence Model

We have already introduced the influence model, where the influence strength matrix $\mathbf{R}$ remains the same for all t. However, there is extensive evidence leading us to think that influence is indeed a dynamical process[21, 22]. This can also be seen from many real-world experiences: Friendship is not static, and the person who currently possesses the most influence over you may be different after some time; In a tedious neogotiation with many parties involved, your most active opponent may change due to topic shift and strategy shift over time... Therefore, we believe that, in a social system such as a group discussion session, the influence between subjects fluctuates as well.

We now demonstrate how the influence model is extended to the dynamical case, and we call this generalization the *Dynamical Influence Model*. Instead of having one single influence strength matrix, $\mathbf{R}$, we here consider a finite set of different influence strength matrices, $\{\mathbf{R}(1), \ldots, \mathbf{R}(J)\}$, each represents a difference influence dynamical pattern between entities. J is a hyper parameter set by users to define the number of different interaction patterns. Our approach is basically a switching model, and we also introduce the switching latent state $r_t \in \{1, \ldots, J\}, t = 1, \ldots, T$, which indicates the current active influence matrix at time t. Therefore, Eq. 2 turns to the following:

$$\text{Prob}(h_t^{(c')}|h_{t-1}^{(1)}, \ldots, h_{t-1}^{(C)}) = \sum_{c \in \{1,\ldots,C\}} \mathbf{R}(r_t)_{c',c} \times \text{Prob}(h_t^{(c')}|h_{t-1}^{(c)}). \tag{3}$$

As r_t switches to different values between 1 to J at different time t, the dynamics at different time t is then determined by different influence matrices $\mathbf{R}(r_t)$.

As shown in Section 6.1, we realize that it is very important to constrain the switching of r_t for two reasons: a) In many social systems, the change of influence patterns changes slowly and gradually. b) A prior eliminates the probability of overfitting. Therefore we introduce the following prior for r_t:

$$r_{t+1}|r_t \sim \text{multi}(V_{r_t,1}, \cdots, V_{r_t,J}), \tag{4}$$

where $\mathbf{V}$ is a system parameter matrix constrained by another hyper-parameter $p^V, p^V > 0$. The prior is shown in Eq. 5.

$$(V_{r_t,1},\ldots,V_{r_t,J}) \sim \text{Dirichlet}(\underset{\substack{\uparrow\\1,}}{10^0}, \underset{\substack{\uparrow\\2,}}{10^0}, \underset{\ldots,}{\ldots}, \underset{\substack{\uparrow\\r_t,}}{10^{p^V}}, \underset{\ldots,}{\ldots}, \underset{\substack{\uparrow\\J}}{10^0}). \tag{5}$$

This prior provides a better control of the process $r_1,\ldots,r_T$. If p^V is extremely large, the prior will force r_{t-1} and r_t to be same; If p^V is very small, the prior will turns to a uniform distribution and allow r_{t-1} to randomly switch to any value in $\{1, ..., J\}$ with equal probability.

Given the model description and hyper parameters J and p^V, we can then write the likelihood function:

$$\mathcal{L}(O_{1:T}^{1:C}, h_{1:T}^{1:C}, r_{1:T}|\mathbf{E}^{1:C}, \mathbf{F}^{1:C}, \mathbf{R}(1:J), \mathbf{V}) \tag{6}$$

$$= \prod_{t=2}^{T}\Big\{\text{Prob}(r_t|r_{t-1}) \times \prod_{c=1}^{C}\Big[\text{Prob}(O_t^{(c)}|h_t^{(c)}) \times \text{Prob}(h_t^{(c)}|h_{t-1}^{(1,\ldots,C)}, r_t)\Big]\Big\}$$

$$\times \prod_{c=1}^{C}\text{Prob}(O_1^{(c)}|h_1^{(c)})\text{Prob}(h_1^{(c)})\text{Prob}(r_1). \tag{7}$$

To better understand how we extend the influence model into the dynamical influence model, we illustrate the Bayesian graph for both models in Fig. 2.

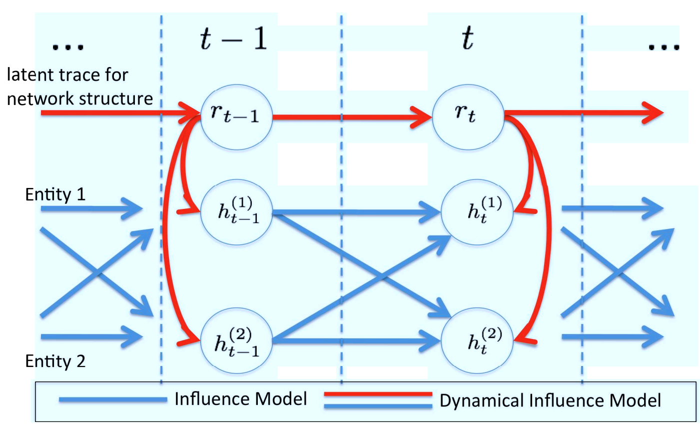


Figure 2: A graphical representation of our model when $C = 2$. The blue lines constitute the dependence of the influence model described in Section 2. The red line is the additional layer which brings additional switching capacity to the influence model, and together with blue lines constitute the variable dependence for the dynamical influence model.

4.1 Review in Dynamical Networks

There are quite a few related projects if we consider influence dynamics as the dynamics in network structure. Researchers have been studying the class of time-varying network models: from EGRM [23] to TESLA [24], which model network rewiring from correlations in observations. Compared with these models, the dynamical influence model serves as a unique generative approach for modeling noisy signals from a dynamical network. One latest work tends to learn network topology from node state dependence [25], with a binary and static underlying model.

5 Inference

In signal processing applications, we are given the observation time series signals $O_{1:T}^{(1)}, ..., O_{1:T}^{(C)}$, and based on observations we need to learn the distributions for underlying latent variables and the system parameters for dynamical influence model. The inference process for our model is discussed here. Since the dynamical influence model is a generalization of the influence model, the following content suits both models.

In literature on the inference for the influence model class, researchers started with a standard exact inference algorithm (Junction Tree) with exponential complexity, and then moved to an approach based on optimization[12]. Other scholars gradually moved to an approximation approach based on the Forward-Backward algorithm and variational-EM [26, 16]. The influence model can also be trained via other approximation such as the mean field method[27].

Here we show some key steps for the variational E-M approach, which has been developed and applied successfully in many datasets. We refer to readers to Pan et al [16] for detail. Definition is denoted by $\equiv$, and $\sim$ denotes the same distribution but the right side should be normalized accordingly.

5.1 E-Step

We adopt a procedure similar to the forward-backward procedure in HMM literature. First, we define the following forward parameters for $t = 1, ..., T$.:

$$\alpha_{t,c}^{r_t} \equiv \text{Prob}(h_t^{(c)}|r_t, O_{1:t}), \tag{8}$$

$$\kappa_t \equiv \text{Prob}(r_t|O_{1:t}), \tag{9}$$

where $O_{1:t}$ denotes $\{O_{t'}^{(c)}\}_{t'=1,...,t}^{c=1,...,C}$. However, complexity for computing $\alpha_{t_c}^{r_t}$ given $\alpha_{t-1,c}^{r_{t-1}}$ grows exponentially with respect to C, so we adopt the variational approach[28], and E-M is still guaranteed to converge under variational approximation[28]. We proceed to decouple the chains by:

$$\text{Prob}(h_t^{(1)}, ..., h_t^{(C)}|O_{1:t}, r_t) \approx \prod_c Q(h_t^{(c)}|O_{1:t}, r_t), \tag{10}$$

and naturally:

$$\alpha_{t,c}^{r_t} \approx Q(h_t^{(c)}|O_{1:t}, r_t) \tag{11}$$

The approximation adopted here enables us to run inference in polynomial time. Based on this approximation, starting with $\alpha_{1,c}^{r_1}$ and κ_1, we can compute $\alpha_{t,c}^{r_t}$ and κ_t, $\forall t = 2, ..., T$ step by step in a forward manner.

Using the same idea, we can compute the following backward parameters for all t in the backward order (i.e. start with $t = T$, then compute $\beta_{t,c}^{r_t}$ and ν_t for $t = T-1, T-2, ..., 1$):

$$\beta_{t,c}^{r_t} \equiv \text{Prob}(h_t^{(c)}|r_t, O_{t:T}), \tag{12}$$

$$\nu_t \equiv \text{Prob}(r_t|O_{t:T}). \tag{13}$$

5.2 M-step

With κ_t and ν_t, we can estimate:

$$\xi_{i,j}^t \equiv \text{Prob}(r_t = i, r_{t+1} = j | O_{1:T}) = \\ \text{Prob}(r_t = i | O_{1:t}) \text{Prob}(r_{t+1} = j | O_{t+1:T}) \text{Prob}(r_{t+1} | r_t) / \\ \sum_{i,j} \text{Prob}(r_t = i | O_{1:t}) \text{Prob}(r_{t+1} = j | O_{t+1:T}) \text{Prob}(r_{t+1} | r_t), \tag{14}$$

$$\lambda_i^t = \text{Prob}(r_t = i | O_{1:T}) = \frac{\sum_j \xi_{i,j}^t}{\sum_i \sum_j \xi_{i,j}^t}, \tag{15}$$

and update V by:

$$\mathbf{V}_{i,j} \leftarrow \frac{\sum_t \xi_{i,j}^t + k}{\sum_t \sum_j \xi_{i,j}^t + p^V}, \tag{16}$$

where $k = p^V$ if $i = j$, 0 otherwise.

We then compute the joint distribution $\text{Prob}(h_t^{q_{t+1}^{(c)}}, h_{t+1}^{(c)}, r_{t+1} | O_{1:T})$, and update parameters such as influence matrices $\mathbf{R}(1), ..., \mathbf{R}(J)$, $\mathbf{E}^c$ and $\mathbf{F}^c$ by marginalizing this joint distribution.

6 Application

We now explain how we can use our model in signal processing.

6.1 Toy Example: Two Interacting Agents

In this toy example, we demonstrate how the dynamical influence model can be applied to find structural changes in network dynamics. As a tutorial, we also explain how readers should adjust two hyper parameters J and p^V in using this model.

From a dynamical influence process composed of two interacting entities, we sample two binary time series of 600 steps. Each chain has two hidden states with a random transition biased to remaining in the current state. We sample binary observations from a random generated multinomial distribution. To simulate a switch in influence dynamics, we sample with influence matrix $\mathbf{R}(1)$ (shown in Table 1) in the first 200 frames, and later on we sample with influence matrix $\mathbf{R}(2)$. We purposely make the two configuration matrices different from each other. Partial data are shown in Table 1 (left). We use the algorithm in Section 5 to infer the dynamical influence model's parameters $\mathbf{V}, \mathbf{R}(1:J), \mathbf{E}^{1:C}, \mathbf{F}^{1:C}$. All parameters (including the emission distribution) are initialized randomly, and they are estimated automatically during the E-M process.

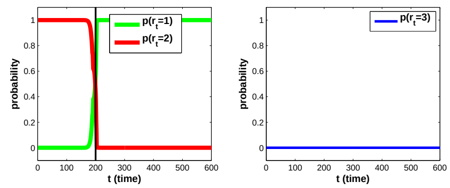
Table 1: Left: Part of the two input toy sequences for a two-chain dynamical influence process. Right: The original two influence matrices of the toy model and the same matrices learned by our algorithm with $J = 3$ and $p^V = 10^1$.

SEQ. NO.	DATA(PARTIALLY)
1	221111121212212...
2	112111212121122...

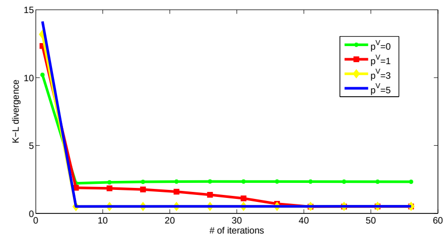
	$\mathbf{R}(1)$	$\mathbf{R}(2)$
True	$\begin{pmatrix} 0.90 & 0.10 \\ 0.10 & 0.90 \end{pmatrix}$	$\begin{pmatrix} 0.05 & 0.95 \\ 0.95 & 0.05 \end{pmatrix}$
Learned	$\begin{pmatrix} 0.93 & 0.07 \\ 0.10 & 0.89 \end{pmatrix}$	$\begin{pmatrix} 0.08 & 0.92 \\ 0.94 & 0.06 \end{pmatrix}$

Choosing Hyper-Parameters: We now discuss the selection of hyper-parameters J and p^V for the dynamical influence model. For the number of active influence matrices J, we illustrate its characteristics by running the same example with $J = 3$. We show the poster distribution of r_t (calculated in Eq. 15) in Fig. 3(a). The dynamical influence model discovers the sudden change of influence weights accurately at $t = 200$. Since the toy process only has two true configuration matrices, the posterior probability of the 3rd configuration being active is almost zero for any t. The system properties are fully captured by the other two configuration matrices during the training. The learned configuration matrices (shown in Table 1) are correctly recovered. Based on Fig. 3(a) and experiments with other values for J (which we can not show here due to the space limitation), we suggest that readers should gradually increase J until the newly added configuration matrices are no longer useful in capturing additional dynamical information from the data, by ensuring there is no constant zero posterior probability as in the right plot in Fig. 3(a).

We also demonstrate convergence of the K-L Divergence between the true distributions of the transition probability and the learned distributions in Fig. 3(b) with different values of p^V. As can be seen in Fig. 3(b), the algorithm converges quickly within 50 iterations. However, when p^V is small, we may encounter over-fitting where the learned model rapidly switches between different configurations to best suit the data. Therefore, in Fig. 3(b), the divergence for $p^V = 0$ remains higher than other p^V values at convergence. In conclusion, we advise users to increase p^V gradually



(a)



(b)

Figure 3: (a): The posterior of r_t is shown with $J = 3$ after convergence. The middle black vertical line on the left indicates the true switch in r_t. The probability of $\mathbf{R}(1)$ being active and $\mathbf{R}(2)$ being active are shown in the left plot; $\mathbf{R}(3)$ is shown in the right, which remains inactive. (b): The K-L divergence between learned parameters and the true distributions with respect to number of iterations.

until the posterior of r_t does not fluctuate.

6.2 Modeling Dynamical Influence in Group Discussions

6.2.1 Dataset Description and Preprocessing

Researchers in [29] recruited 40 groups with four subjects in each group for this experiment. During the experiment, each subject was required to wear the sociometric badge on their necks for audio recording(see the right picture in Fig. 1), and each group was required to perform two different group discussion tasks: a brainstorming task and a problem solving task. Each task usually lasted for 3 to 10 minutes. We kindly refer readers to the original paper[29] for details on data collection and experiment preparations.

The groups were asked to perform these tasks in two different settings: (a) being *co-located* in the same room around a table and (b) being *two pairs* in two rooms with only audio communication available between the pairs. (The badge is deployed in both cases for audio collecting.) Later in the paper we refer to these two settings as CO and DS respectively. We separate all samples into four categories according to their original context and content as explained in Table 2. Since discussions are held in four-person groups, each sample for a discussion session is composed of four sequences collected by the four badges on participants' chests. The audio sequence picked up by each badge is split into one-second blocks. Variances of speech energy are calculated for each block. We then applied a hard threshold to convert them into binary sequences. In all experiments, we only use binary sequences as data input.

6.2.2 Predicting Turn Taking in Discussion

One important aspect of modeling interaction dynamics is the ability to predict turn taking–who will speak next in the interaction process. We here explain an application of our dynamical influence process to predict turn taking, and we show that it is possible to achieve good accuracy in prediction given only the audio volume variance observations, with no information from the audio content. We consider in this application that influence is the effect of someone speaking on other participants' turn taking behavior.

Ten occurrences of turn taking behavior from each sample are selected for prediction purposes. "Turn taking" here is defined as incidences in which the current speaker ceases speaking, and another speaker starts to speak.

For the dynamical influence model, we model each person as an entity c, and the observed audio variances at time t as $O_t^{(c)}$. Each person also two hidden states, representing if the person is speaking or not speaking. The hidden layer eliminates error due to noise and non-voicing speaking in audio signals[30]. Therefore, influence here is set to capture how each person's speaking/non-speaking hidden states dynamically changes other person's speaking/non-speaking states,i.e., how people influence each others' turn taking.

All parameters are initialized randomly and learned by the E-M inference algorithm in this example. Since our algorithm is a generative process, we sample time t from our model, and mark the chain that changes the most toward the high-variance observations as the turn taker. The emission probability $\text{Prob}(O_t^{(c)}|h_t^{(c)})$ is modeled using a multinomial distribution, and is estimated automatically during the E-M process.

To compare, we also show results using TESLA and nearest neighbors. For TESLA, we use the official implementation[31] to obtain the changing weights between pairs of nodes, and we pick the node which has the strongest correlation weight to other nodes at $t-1$ as the turn taker at t.

Table 2: The description for four different categories of all the samples.

CATEGORY	**TASK DESCRIPTION**
CO+PS	Four people perform a problem solving task in the same room.
CO+BS	Four people perform a brainstorming session in the same room.
DS+PS	Four people perform the same problem solving task in two rooms with Skype.
DS+BS	Four people perform the same brainstorming session in two rooms with Skype.

To predict the turn taking at time t using the nearest neighbor method, we look over all previous instances of turn taking behaviors that have the same speaker as the one in $t-1$, and predict by using the most frequent outcomes.

Table 3: Accuracy for different turn taking prediction methods on both the full dataset and the half of the dataset with more complex interactions. The random guess accuracy is 33%. Human accuracy is typically around 50% for similar tasks[32].

	ACCURACY ALL SAMPLES				**ACCURACY COMPLEX INTERACTION SAMPLES**			
METHODS	**DS+BS**	**DS+PS**	**CO+BS**	**CO+PS**	**DS+BS**	**DS+PS**	**CO+BS**	**CO+PS**
TESLA	0.41	0.42	0.32	0.25	0.44	0.37	0.37	0.17
NN	**0.58**	0.60	0.48	0.50	**0.47**	0.47	0.38	0.26
Ours(J=1)	0.45	**0.67**	**0.75**	**0.63**	0.45	0.56	**0.77**	0.62
Ours(J=2)	0.46	0.58	0.65	0.34	**0.47**	0.58	0.67	0.46
Ours(J=3)	0.50	0.60	0.55	0.48	**0.47**	**0.73**	0.65	**0.65**

The accuracy for each algorithm is listed in Table 3. We also show the prediction accuracy for the half of all samples that have more complex interactions, i.e., higher entropy. For our dynamical influence based approach, we list error rates for $J = 1, 2$ and 3. Except DS+BS, We notice that the dynamical influence model outperforms others in all categories with different J. This performance is quite good considering that we are using only volume and that a human can only predict at around 50% accuracy for similar tasks[32].

More importantly, the dynamical influence model seems to perform much better than the competing methods for more complex interactions. For simple interactions, it seems that $J = 1$ or even NN perform the best due to the fact that there is little shift in influence structure during the discussion. However, when handling complex interaction processes, the introduction of a switching influence dynamics dramatically improves the performance as shown in Table 3. This results suggest that the dynamical influence assumption is reasonable and necessary in modeling complex group dynamics, and it can improve prediction accuracy significantly. However, in simple cases, the model achieves the highest performance only when $J = 1$, i.e. the influence is static, and a higher J will only lead to overfitting.

7 Modeling Flu Epidemics as Influence Dynamics

The last example we want to demonstrate is the flu spreading dynamics. We apply our algorithm to the weekly US flu activity data from Google Flu Trend [33]. All 50 states in US are divided into ten regions by their geo-location, as shown in Fig. 4, and we model each region as an entity in the dynamical influence model.

Figure 4: Ten regions of the United States defined by US Health and Human Services.

As the data is continuous, six hidden states are used for each chain, and $p(O_t^{(c)}|h_t^{(c)})$ is modeled with six different Gaussian distributions with different means and the same variance for each hidden state. We set by hand the six mean values so that they represent the six different severe levels for the flu epidemics, from the least severe to the most severe. We train the model using the first 290 weeks (from 2003 to early 2009), and we show the posterior for r_t, the switching parameter, in Fig.5 together with the three learned influence matrices. While there are many small peaks suggesting changes in influence, the probability changes dramatically around Christmas, which suggests that the influence patterns among these ten regions are very different during the holiday season. *The dynamical influence model actually reveals Christmas traveling by discovering the change in epidemic dynamics*.

Influence matrix 1 captures the dynamics during holiday seasons, while influence matrix 2 captures the dynamics during normal seasons. Row i corresponds to the region i in Fig. 4. Let's take an example of Row 1, the New England region. During normal time as shown in the 1st row of influence matrix 2, New England is more likely to be influenced by close regions such as 3 and

4; during holiday seasons, New England is more likely to be influenced by all regions especially distant regions such as region 9. The same phenomena exist for other regions as well.

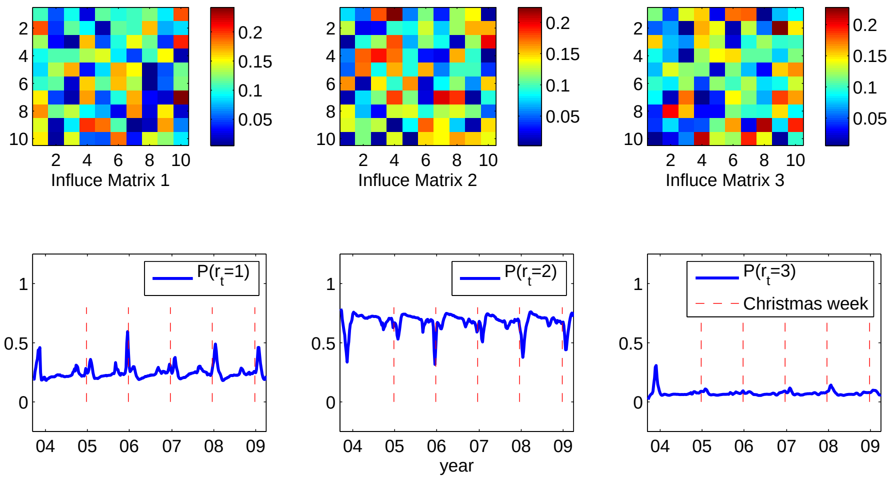


Figure 5: The inferred posterior for r_t given all observations after convergence is shown here. While there are many small peaks indicating changes in influence, the largest peaks occur at Christmas holiday seasons, which implies that influence between states are very different in Christmas holidays comparing with other dates. This matches the common sense that traveling patterns are different around holiday seasons. In our experiment, we find that three configuration matrices are good enough to capture the flu dynamics.

8 Discussions

We introduce the influence model and its generalization, the dynamical influence model in this article. We explain the model definition, model training, and different applications in processing social signals.

The most interesting aspect of research on the influence model is that the influence matrix $\mathbf{R}$ connects the social network and the stochastic process of state transition. The switching matrices $\mathbf{R}(1), ..., \mathbf{R}(J)$ are even able to bridge state transition to time-varying networks. In addition, the weighted conditional probability in Eq. 2 is very similar to the social norm transmission on social networks[4]. All the above properties make the influence model a unique tool for social computing.

However, the influence model shares the same issue with other machine learning models: the inference requires sufficient training data, and tuning is necessary for best results. Also, the influence model learns the influence network topology from data, and many existing social relationship information is not properly utilized by this model. Immediate future works are to leverage existing network data in the influence model, and to study its performance with limited data series.

Influence remains an intriguing research focus in computational social science. In many scenarios for social data processing, the definition of "influence" in the influence model may not be adequate. Researchers are encouraged to carefully consider their data and choose the right approach.

Wei Pan (panwei@media.mit.edu) Wei Pan is a Ph.D student in the Human Dynamics Group at MIT Media Lab. He obtained his B.Eng. degree in Computer Science from Tsinghua University in Beijing, China. He has also worked for Google R&D China for a year and a couple of start-ups.

Wen Dong (wdong@media.mit.edu) Wen Dong is a post-doctoral research associate at the MIT Media Lab. Most of his research centers around using the network dynamics theory to model the collective intelligence. Wen Dong has a Ph.D. degree and a M.S.degree from the MIT Media Lab, M.S. degrees in CS and nuclear engineering from the University of San Francisco and Tsinghua University respectively, and a B.E. degree in CS from Xi'an Jiaotong University.

Manuel Cebrian (cebrian@mit.edu) Manuel Cebrian is a Postdoctoral Associate with MIT'S Media Laboratory. Previously, Manuel worked at Telefonica Research (2009-2008), and Brown University (2008-2007). He holds a PhD in Computer Science from Universidad Autonoma de Madrid (2007).

Taemie Kim (taemie@media.mit.edu) is a PhD candidate at the MIT Media Lab. Her research focuses on understanding and improving distributed collaboration using sociomteric sensors. Her research is centered around Computer Supported Collaborative Work, utilizing disciplines of interaction design, machine learning, organizational behavior, and social psychology. Prior to MIT, she earned a B.S. in EECS from KAIST, Korea and a M.S in EE from Stanford University.

Alex "Sandy" Pentland (pentland@media.mit.edu) Professor Alex "Sandy" Pentland is a pioneer in organizational engineering, mobile information systems, and computational social science. He directs the MIT Human Dynamics Lab, and the Media Lab Entrepreneurship Program, including the EPROM entrepreneurship program in Africa. He is among the most-cited computer scientists in the world. His most recent book, Honest Signals: How They Shape Our World was published in 2008 by the MIT Press.

References

[1] E. Katz and P.F. Lazarsfeld. *Personal influence*. Free Pr., 1955.

[2] D.J. Watts and P.S. Dodds. Influentials, networks, and public opinion formation. *Journal of Consumer Research*, 34(4):441–458, 2007.

[3] A.W. Woolley, C.F. Chabris, A. Pentland, N. Hashmi, and T.W. Malone. Evidence for a collective intelligence factor in the performance of human groups. *science*, 330(6004):686, 2010.

[4] S. INFLUENCE. Social norms, conformity, and compliance. *The handbook of social psychology*, page 151, 1998.

[5] G.F.. Farris and F. Lim. Effect of performance on leadership, cohesiveness, influence, satisfaction and subsequent performance. *Journal of Applied Psychology*, 53(6):490–497, 1969.

[6] Daniel. Olguin Olguin, Peter A. Gloor, and Alex. Pentland. Capturing individual and group behavior with wearable sensors. In *AAAI Spring Symposium on Human Behavior Modeling*, Stanford, CA, March 2009. AAAI.

[7] N. Eagle and A. Pentland. Reality mining: sensing complex social systems. *Personal and Ubiquitous Computing*, 10(4):255–268, 2006.

[8] C.M. Bishop et al. *Pattern recognition and machine learning*. Springer New York:, 2006.

[9] M. Brand, N. Oliver, and A. Pentland. Coupled hidden Markov models for complex action recognition. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 994–999, 1997.

[10] W. Dong and A. Pentland. Modeling influence between experts. *Lecture Notes in Computer Science*, 4451:170, 2007.

[11] A. Madan and A. Pentland. Modeling Social Diffusion Phenomena using Reality Mining. In *AAAI Spring Symposium on Human Behavior Modeling. Palo Alto, CA*, 2009.

[12] T. Choudhury and S. Basu. Modeling conversational dynamics as a mixed memory markov process. In *Proc. of Intl. Conference on Neural Information and Processing Systems (NIPS)*. Citeseer, 2004.

[13] S. Basu, T. Choudhury, B. Clarkson, A. Pentland, et al. Learning human interactions with the influence model. *MIT Media Laboratory Technical Note*, 2001.

[14] W. Dong, B. Lepri, A. Cappelletti, A.S. Pentland, F. Pianesi, and M. Zancanaro. Using the influence model to recognize functional roles in meetings. In *Proceedings of the 9th international conference on Multimodal interfaces*, pages 271–278. ACM, 2007.

[15] W. Dong and A. Pentland. A Network Analysis of Road Traffic with Vehicle Tracking Data. 2009.

[16] W. Pan, M. Cebrian, W. Dong, T. Kim, and A. Pentland. Modeling Dynamical Influence in Human Interaction Patterns. *Arxiv preprint arXiv:1009.0240*, 2010.

[17] D. Zhang, D. Gatica-Perez, S. Bengio, and D. Roy. Learning Influence Among Interacting Markov Chains. 2005.

[18] E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky. Nonparametric Bayesian learning of switching linear dynamical systems. In *Neural Information Processing Systems 21*. MIT Press, 2009.

[19] W. Pan and L. Torresani. Unsupervised hierarchical modeling of locomotion styles. In *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM New York, NY, USA, 2009.

[20] C. Castellano and V. Loreto. Statistical physics of social dynamics. *Reviews of Modern Physics*, 81(2):591, 2009.

[21] S.A. Ansari, V.S. Springthorpe, and S.A. Sattar. Survival and vehicular spread of human rotaviruses: possible relation to seasonality of outbreaks. *Reviews of infectious diseases*, 13(3):448–461, 1991.

[22] J.P. Onnela, J. Saramaki, J. Hyvonen, G. Szabó, D. Lazer, K. Kaski, J. Kertész, and A.L. Barabási. Structure and tie strengths in mobile communication networks. *Proceedings of the National Academy of Sciences*, 104(18):7332, 2007.

[23] F. Guo, S. Hanneke, W. Fu, and E.P. Xing. Recovering temporally rewiring networks: A model-based approach. In *Proceedings of the 24th international conference on Machine learning*, page 328. ACM, 2007.

[24] A. Ahmed and E.P. Xing. Recovering time-varying networks of dependencies in social and biological studies. *Proceedings of the National Academy of Sciences*, 106(29):11878, 2009.

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

[26] W. Dong and A. Pentland. Multi-sensor data fusion using the influence model. In *Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks*, page 75. Citeseer, 2006.

[27] Dong, Wen. *Modeling the Structure of Collective Intelligence*. PhD thesis, Massachusetts Institute of Technology, 2010.

[28] M.I. Jordan, Z. Ghahramani, T.S. Jaakkola, and L.K. Saul. An introduction to variational methods for graphical models. *Machine learning*, 37(2):183–233, 1999.

[29] T. Kim, A. Chang, L. Holland, and A.S. Pentland. Meeting mediator: enhancing group collaborationusing sociometric feedback. In *Proceedings of the ACM 2008 conference on Computer supported cooperative work*, pages 457–466. ACM, 2008.

[30] S. Basu. A linked-HMM model for robust voicing and speech detection. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, volume 1, pages I–816. IEEE, 2003.

[31] A. Ahmed and E.P. Xing. Recovering time-varying networks of dependencies in social and biological studies. *Proceedings of the National Academy of Sciences*, 106(29):11878, 2009.

[32] D. Schlangen. From reaction to prediction: Experiments with computational models of turn-taking. In *Ninth International Conference on Spoken Language Processing*. Citeseer, 2006.

[33] J. Ginsberg, M.H. Mohebbi, R.S. Patel, L. Brammer, M.S. Smolinski, and L. Brilliant. Detecting influenza epidemics using search engine query data. *Nature*, 457(7232):1012–1014, 2008.