Modeling adoptions and the stages of the diffusion of innovations

Yasir Mehmood
Pompeu Fabra University, Spain
yasir@yahoo-inc.com

Nicola Barbieri
Yahoo Labs, Barcelona, Spain
barbieri@yahoo-inc.com

Francesco Bonchi
Yahoo Labs, Barcelona, Spain
bonchi@yahoo-inc.com

***Abstract*—We study the data mining problem of modeling adoptions and the stages of the diffusion of an innovation. For our aim we propose a stochastic model which decomposes a diffusion trace (sequence of adoptions) in an ordered sequence of stages, where each stage is intuitively built around two dimensions: users and relative speed at which adoptions happen. Each stage is characterized by a specific rate of adoption and it involves different users to different extent, while the sequentiality in the diffusion is guaranteed by constraining the transition probabilities among stages.**

An empirical evaluation on synthetic and real-world adoption logs shows the effectiveness of the proposed framework in summarizing the adoption process, enabling several analysis tasks such as the identification of adopter categories, clustering and characterization of diffusion traces, and prediction of which users will adopt an item in the next future.

I. Introduction

The spread of new ideas or technology in a society is a complex process that starting from a small fraction of the population, propagates over time through a diverse set of communication channels, potentially reaching a critical mass. Understanding the dynamics of such complex process is an important task with implications for sociology and economics, as well as important applications, especially in marketing.

The most popular model in this area was proposed by Everett Rogers in his seminal work *"Diffusion of Innovations"*, first published in 1962 [11]. Rogers' theory models the process of diffusion of innovation using four main elements: the innovation, the communication channels, time, and a society. Those elements work in synergy to produce a diffusion: an innovation is communicated through a variety of channels, over time, among the members of a social system. While the term *"diffusion"* refers to the overall process at the level of the social system, the term *"adoption"* refers to the sub-process that brings the single individual to the decision of adopting the innovation. Rogers' theory provides a categorization of individuals, based on their propensity to innovate, in five classes. The individuals who tend to be the first in adopting innovations are labeled as *innovators*. The *early adopters* tend to adopt ideas after innovators and hold leadership roles in the social system. Those are responsible for bringing the innovation to the attention of the mass market. *Early majority* is made of individuals that waits until most of their peers adopt the innovation. *Late majority*, is the part of the population who tend to adopt an innovation after the average member of the society does. Finally *laggards* are the last to adopt an innovation.

In this paper we study the data mining problem of modeling adoptions and the stages of the diffusion of an innovation. Our unique input is a database of adoptions $\mathbb{D}$, which is a relation `(User,Item,Time)` where a tuple $\langle u, i, t\rangle \in \mathbb{D}$ indicates that the user u adopted the item i at time t.

While a unifying one-model-fits-all theory (as the one by Rogers) is appealing, when it comes to modeling real-world data, more flexibility is needed. In fact, real-world items exhibit consistent differences in the way they diffuse. This is particularly true if we consider the speed with which nowadays ideas, news, opinions or rumors can propagate through a variety of new communication channels, such as on-line social networks, microblogs, instant messenger systems, and so on.

Firstly, as users are typically interested in a limited number of topics, the diffusion of different items may interest different segments of the market, e.g. news about finance are unlikely to interest teenagers. Secondly, some items may be widely adopted, whereas others could be adopted by a limited number of individuals. In other terms, the diffusion of a item can achieve different level of success. Finally, different items may exhibit different temporal patterns of diffusion. For instance, the diffusion of news or tweets happens rapidly and fades out in few days, whereas the diffusion of books or movies may last for years after their release.

These considerations motivate the need for a more fine grained analysis of the diffusion process, aimed at detecting different groups of items which share similar diffusion patterns, and for each detected group, capture the main diffusion characteristics (e.g., the stages of diffusions, their temporal length and adoption rate) in a simple and useful abstraction.

A. Contributions and roadmap

The main contribution of this paper is MASD, *a stochastic framework for modeling adoptions and the stages of diffusions*, which realizes a good compromise between descriptive capabilities and simplicity. At the high level, the process of diffusion is decomposed in a finite and ordered sequence of stages of adoptions, where early stages correspond to the introduction in the market of a item, while latter ones correspond to the maturity phase of its life cycle.

The key idea of our framework, is to enforce a 1-to-1 association between the stages of adoptions, and the states

of a Markov model. In particular, in order to capture the natural sequentiality of the stages of diffusion, we use a special type of hidden Markov model, known as *left-to-right*, where the diffusion of an item starts from the first state of the model, progresses through later states, never backtracking to previous states. The number of states is automatically detected by relying on the Bayesian Information Criterion (BIC). Each state involves some users more than others, and it is characterized by a specific rate of adoption, which controls the elapsed time between consecutive adoptions.

In order to better explain the diffusion mechanisms, we devise a learning framework that alternates two phases (i) clustering the diffusions in different groups, and (ii) for each group fits the parameter of the MASD model by means of an Expectation Maximization process.

We present a thorough empirical evaluation using both synthetic and real-world data. Experiments on synthetic data with planted clustering structure confirm the accuracy of the learning framework, which is also shown to scale linearly with the size of data and the number of models. Experiments on real-world diffusion data show interesting patterns of adoption, with large diversity among the clusters produced, in terms of diffusion size and speed. Experiments also shows that a user is, in most of the cases, bound to one or two states maximum.

Having learned a stochastic model allows us to use it for predictive purposes. In our experiments, we show how we can use it to accurately predict, for an on-going diffusion, which are the users that more likely will adopt the item in a future time window (e.g., next week). This capability enables dynamic marketing strategies, in which we focus on targeting specific segments of users, that are likely to adopt the product in the near future.

The rest of the paper is organized as follows. In the next section we provide a brief overview of related literature. In Section III we introduce our MASD model, while in Section IV we present the learning framework. Section V contains our thorough experimentation, and Section VI concludes the paper.

II. Related work

The problem of understanding the dynamics (how, why and at which rate) that characterize the process of diffusion of innovations has received a considerably amount of attention by different research communities (anthropology, geography, economy and sociology, just to name a few). Rogers' seminal work [11] provides a unified tool for modeling diffusion processes and it has inspired, among others, several studies that focus on how information spread on the Web and on online social networks. In fact, the Web provides a very effective communication channel for the diffusion of topics, news and rumors. Leskovec *et al.* [8] propose a framework for tracking short phrases ("memes") across mainstream media sites, and to study the diffusion dynamics of the news cycle. Similar techniques enable the tracking of emerging trends on social and mainstream media [6], [9] and the study of the dynamics of diffusion process across different kinds of topics [12].

A complementary perspective focuses the role of people in the diffusion of information. Rogers' theory identifies five categories of people by considering the adoption time of each person with respect to the rest of the population. Given a log that records browsing behavior, Mele *et al.* [10] tackle the problem of identifying users who discover interesting Web pages before others (early adopters), and such information is exploited for recommendation purposes. Saez *et al.* [13] extend Rogers' categorization of users, by introducing the concept of *trendsetters*, i.e. people that adopt and effectively boost the spread of new ideas before these ideas become popular. Being an early adopter does not imply being a trendsetter, as the latter requires the ability of propagating information to their social peers through word-of-mouth phenomena.

Social networks enable individuals to share information with their social peers; in this context, few, high influential, users can trigger large cascade of adoptions. Bakshy *et al.* [2] study how the social network affects online information diffusion. Their findings confirm that users exposed to the diffusion of an item are significantly more likely to adopt it than those who are not exposed. However, the dynamics of diffusion of information in social networks cannot be exclusively described by local models, in which the likelihood of adoption for one user depends only on the adoptions by his peers. Borrowing Rogers' categories, Budak *et al.* [4] confirm that the likelihood of adoption depends also on the behavior of the considered user with respect to the entire population.

The research contributions summarized above attempt at modeling diffusions as either global or local processes, but they overlook the modeling of the different stages of diffusions, that entails different dynamics and different intensity with which information is adopted. Another main distinction, is that the bulk of this literature focuses on analyzing propagations through a social network, i.e., when the social structure of the population is an input to the problem. In our work instead, we model diffusion in a general population, when no social network is given ($\mathbb{D}$ is the only input to our problem).

A setting closer to ours is that of finding bursts in information streams. In [7] Kleinberg models the burst of activity in a data stream as a probabilistic automaton, where each state is associated with a different level of burstiness. The evolution of the inter-arrival times between observations in the stream is captured by state transitions in a specified a hierarchical structure, where each state has a level of burstiness higher than the previous one. Finally, the analysis temporal patterns in the adoption of online content can be cast as time series clustering problem [15], where each cluster is characterized by a distinct shape of popularity across time.

III. Modeling the stages of diffusion

We are given an *adoption log* $\mathbb{D}$, which is a relation `(User,Item,Time)` where a tuple $\langle u, i, t\rangle \in \mathbb{D}$ indicates that the user u adopted the item i at time t. This data is the unique input to our problem.

Let $\mathcal{U} = \{u_1, \cdots, u_M\}$ and $\mathcal{I} = \{i_1, \cdots, i_N\}$ denote the user-set and items-set, respectively. Each diffusion trace $\mathbb{D}_i$ is

TABLE I: *Main notation used.*

Symbol	Description	Symbol	Description
$\mathbb{D}$	adoption log	$\|\mathbb{D}\|$	total number of adoptions
$\mathcal{U}$	user-set	M	number of users
$\mathcal{I}$	item-set	N	number of items
$\mathbb{D}_i$	adoption trace for item i	$\|\mathbb{D}_i\|$	number of adoptions in $\mathbb{D}_i$
$u_{i,n}$	n-th user adopting i	$t_{i,n}$	time of the n-th adoption of i
s_j	j-th state	K	number of states
$\phi_{u,j}$	$P(u\|\phi_j)$	λ_j	adoption rate in the state s_j
$a_{j,k}$	transition probability from state j to state k	$q_{i,n}$	$\langle u_{i,n}, t_{i,n}\rangle$

a record of individuals and their corresponding adoption times of the item i, such that $\mathbb{D}_i = \{\langle u, t\rangle \,|\, \langle u, i, t\rangle \in \mathbb{D}\}$. We also denote $q_{i,n} = \langle u_{i,n}, t_{i,n}\rangle$ the n-th adoption of item i.

Our goal is to decompose the process of diffusion of items as a finite and ordered sequence of stages, such that given a sequence of past adoptions the current stage is univocally identified. In continuity with Rogers' theory, users have different likelihood of being involved in each stage. Each stage is further characterized by a rate which describes the relative speed of adoption. These concepts can be formalized by properly instantiating the density function for observing a specific adoption given all the previous ones and the current status of the process (i.e., the current stage s_j).

We formulate such density function as:

$$f(\langle u_{i,n}, t_{i,n}\rangle \,|\, \langle u_{i,n-1}, t_{i,n-1}\rangle, \cdots, \langle u_{i,1}, t_{i,1}\rangle, s_j; \Theta) = P(u_{i,n}|\Theta_j) \cdot f(t_{i,n}|t_{i,n-1}, \Theta_j) = \phi_{u_{i,n},j} \cdot f(t_{i,n}|t_{i,n-1}, \lambda_j). \quad (1)$$

In Equation 1, $\Theta = (\Phi, \Lambda)$ represents the set of parameters of the model: (i) Φ is a $M \times K$ matrix, where ϕ_j is a multinomial distribution over users for the stage j and $\phi_{u,j}$ is the probability that the individual u will adopt the innovation in the stage j; (ii) Λ is a K-vectors that specifies stages-specific adoption rates λ_j; (iii) Θ_j represents the set of parameters governing the j-th stage. Equation 1 entails two assumptions. First, the density function for an adoption $\langle u_{i,n}, t_{i,n}\rangle$ in the stage s_j is defined as the product of observing the adopter $u_{i,n}$ and the probability density function of observing the adoption occurring at time $t_{i,n}$. Second, the probability of observing the adopter $u_{i,n}$ only depends on her propensity $\phi_{u_{i,n},j}$ of adoption in the given stage s_j, independently of other users that have adopted i so far.

One of the most natural ways of modeling temporal dynamics is to assume that observations occur continuously and independently at a constant rate. In our model the adoption time only depends on the time of the previous adoption and the stage-specific adoption rate λ_j. This memoryless process can be described by employing an exponential distribution: during each stage s_j, temporal gaps between consecutive adoptions are independent and identically distributed according the following density function:

$$f(t_{i,n}|t_{i,n-1}, \lambda_j) = \lambda_j \exp\{-\lambda_j \cdot \delta_{i,n}\},$$

where $\delta_{i,n} = t_{i,n} - t_{i,n-1}$ is the temporal gap between the n-th adoption and the previous one. The choice of the exponential distribution implies that the expected elapsed time between two consecutive adoptions during stage s_i is $\frac{1}{\lambda_j}$.

We are now ready to introduce our MASD model. The key idea is to consider a 1-to-1 association between the stages of adoptions, and the states of a Markov model.[1] Given K states $\mathbf{S} = \{s_1, \cdots, s_K\}$ (with $K \geq 2$), a Markov model is defined by a transition probability matrix $\boldsymbol{A}$, where $a_{i,j}$ encodes the probability of transition $s_i \to s_j$, i.e., the probability that the next state will be s_j given that the current state is s_i. For all $s_i \in \mathbf{S}$, we have $\sum_{s_j \in \mathbf{S}} a_{i,j} = 1$.

Let $\mathcal{S}(q_{i,n}) \to [1, \cdots, K]$ associate a stage to each observed adoption. In order to enforce a clear sequentiality in the evolution of the stages of adoption we introduce the following constraints:

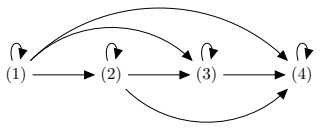
$$\mathcal{S}(q_{i,1}) = s_1, \qquad \mathcal{S}(q_{i,n}) \leq \mathcal{S}(q_{i,n+1}). \quad (2)$$

That is to say that the diffusion of each item i starts from the first state, and after each adoption it can only stay in the current state, or progress through later states, never backtracking to a state that has been left. These structural constraints can be accommodated in a special type of *hidden Markov model (HMM)*, known as *left-to-right*, or *Bakis*, model [1].

A HMM is a Markov model, which at each state outputs observations x. These observation are considered as, either continuous or discrete, *observed* random variables, while the states are *hidden*. The emission of observed variables is governed by state-specific distributions $P(x_n|z_n, \Phi)$, where x_n is the n-th observation, z_n encodes the status of corresponding latent variable, and Φ is a set of parameters governing such distributions. The starting state is defined by a distribution $\Pi = \{\pi_i, \cdots, \pi_K\}, \sum_{j=1}^{K} \pi_j = 1$.

In our setting, the emission probability density for each observation $\langle u_{i,n}, t_{i,n}\rangle$ at the state s_j is given in Equation 1. The main notation we will use in the rest of the paper is briefly summarized in Table I.

In a *left-to-right HMM* the state sequence is such that, as time increases, the state index can only increase, or stay the same. An example of such a model is given in Figure 1: states proceed from left to right (hence the name) and this behavior is achieved by imposing constraints on the values of transition matrix $\mathbf{A}$ ($a_{j,k} = 0$, if $k > j$) and by constraining each sequence to start in the first state (the initial state probability π_j is set to 1 if $j = 1$, zero otherwise).



$$\boldsymbol{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} \\ 0 & a_{2,2} & a_{2,3} & a_{2,4} \\ 0 & 0 & a_{3,3} & a_{3,4} \\ 0 & 0 & 0 & a_{4,4} \end{bmatrix}$$

Fig. 1: *A 4-states left-right HMM and its transition matrix.*

[1]All over the paper we use the term "stage" when referring to the phenomenon we want to model (e.g., "stage of diffusion"), while we use "state" when referring to the concrete model (e.g., "the state of the HMM).

The choice of modeling the likelihood of user's adoption by employing a stage specific multinomial distribution, makes it possible, in principle, to generate multiple times the adoption of the same item by the same user. If we want to avoid this, we can adopt a multivariate hypergeometric distribution, which simulates the sampling from a finite set of element without replacing. However, the multinomial distribution converges to the multivariate hypergeometric distribution for a large size of the sampling population, as it is in our context where $\mathcal{U}$ is supposedly large.

Finally, note that each individual $u \in \mathcal{U}$ is not bound to a unique state, but instead it a distribution of probability of adoption over the set of states. Nevertheless our model allows the identification of the different roles played by users in the diffusion of innovations. If our user u has larger $\phi_{u,j}$ for smaller j, then she can considered an innovator, whereas u can be considered a laggard if $\phi_{u,j}$ is large for large j (i.e., u has more probability of adoptions in the later stages of a diffusion).

In the next section we present an Expectation Maximization (EM) process to learn the parameters of our MASD model from a diffusion log $\mathbb{D}$. As previously discussed in Section I, we do not attempt to model all the diffusions in $\mathbb{D}$ with a unique single MASD model. On the contrary, we propose a learning framework that while fitting the parameters of the model, divide the diffusions in different groups, and produces a MASD model for each group.

IV. Learning

Following the standard EM notation, $\hat{\Theta}$ will represent the current estimate of the set of parameters $\Theta = (\mathbf{A}, \Phi, \Lambda)$. Assuming that each diffusion trace is independent from others, the likelihood of the data given the model parameters Θ, can be expressed as:

$$\mathcal{L}(\Theta; \mathbb{D}) = \sum_{i \in \mathcal{I}} \log \mathcal{L}(\Theta; \mathbb{D}_i). \tag{3}$$

Let $z_{i,n,j}$ be a binary latent variable which is 1 if the n-th adoption in the trace $\mathbb{D}_i$ is associated to the state j, zero otherwise. By assuming that each diffusion trace is independent from others, we can formalize the *Complete-Data Expectation Likelihood* [5] of the generative process, given graphically in Figure 2, as follows:

$$\mathcal{Q}(\Theta, \hat{\Theta}) = \sum_{i \in \mathcal{I}} \left(\sum_{n=2}^{|\mathbb{D}_i|} \sum_{j=1}^{K} \sum_{k \geq j}^{K} \epsilon(z_{i,n-1,j}, z_{i,n,k}) \log a_{j,k} + \sum_{n=1}^{|\mathbb{D}_i|} \sum_{j=1}^{K} \gamma(z_{i,n,j}) \log P(u_{i,n} | \phi_j) + \sum_{n=2}^{|\mathbb{D}_i|} \sum_{j=1}^{K} \gamma(z_{i,n,j}) \log f(t_{i,n} | t_{i,n-1}, \lambda_j) \right),$$

where:

- $\epsilon(z_{i,n-1,j}, z_{i,n,k}) = P(z_{i,n-1,j}, z_{i,n,k} | \mathbb{D}_i, \hat{\Theta})$,
- $\gamma(z_{i,n,j}) = P(z_{i,n,j} = 1 | \mathbb{D}_i, \hat{\Theta})$, i.e., the conditional probability of observing state j for the n-th adoption in the diffusion trace $\mathbb{D}_i$.

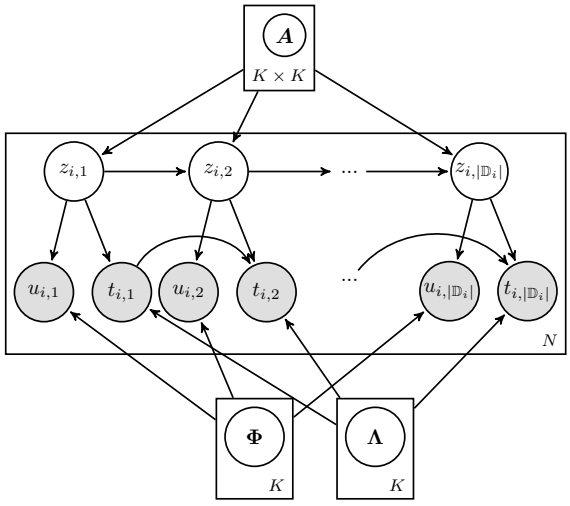


Fig. 2: *Graphical representation of conditional dependencies among observed and latent variables (and their respective states) in MASD.*

Given an initial setting of the parameters $\Theta = (\mathbf{A}, \Phi, \Lambda)$ and we iteratively alternate between the expectation and the maximization learning steps until we meet a chosen convergence criterion (see [3, Chapter 13] for further details).

During the expectation step we compute:

$$\gamma(z_{i,n,j}) = \frac{\alpha(z_{i,n,j})\beta(z_{i,n,j})}{\sum_{k=1}^{K} \alpha(z_{i,n,k})\beta(z_{i,n,k})}$$

$$\epsilon(z_{i,n-1,j}, z_{i,n,k}) = \frac{\alpha(z_{i,n-1,j}) a_{j,k} \beta(z_{i,n,k}) f(q_{i,n} | t_{i,n-1}, \Theta_j)}{\alpha(z_{i,n-1,j})\beta(z_{i,n,j})},$$

where the variables $\alpha(z_{i,n,j}) = f(q_{i,1}, \cdots, q_{i,n}, z_{i,n,j})$ and $\beta(z_{i,n,j}) = f(q_{i,n+1}, \cdots, q_{i,|\mathbb{D}_i|} | z_{i,n,j})$ can be computed efficiently by adopting the *Forward-backward* algorithm [3, pages 618–622], with the emission probability density instantiated as in Equation 1.

In the maximization step we update the parameters of the MASD model as follows:

$$a_{j,k} = \frac{\sum_{i=1}^{N} \sum_{n=2}^{n_i} \epsilon(z_{i,n-1,j}, z_{i,n,k})}{\sum_{i=1}^{N} \sum_{n=2}^{n_i} \sum_{k'=1}^{K} \epsilon(z_{i,n-1,j}, z_{i,n,k'})}$$

$$\phi_{u,j} \propto \sum_{i=1}^{N} \sum_{\substack{n=1 \\ u_{i,n}=u}}^{|\mathbb{D}_i|} \gamma(z_{i,n,j}) \qquad (\sum_{u} \phi_{u,j} = 1,\ 1 \leq j \leq K)$$

$$\lambda_j = \frac{\sum_{i=1}^{N} \sum_{n=2}^{|\mathbb{D}_i|} \gamma(z_{i,n,j})}{\sum_{i=1}^{N} \sum_{n=2}^{|\mathbb{D}_i|} \gamma(z_{i,n,j})\ \delta_{i,n}}.$$

The MASD model that maximizes the likelihood over the

adoption log $\mathbb{D}$, describes in a compact way the diffusion patterns in the underlying data. However, as we expect in the data to coexist very diverse types of diffusions (e.g., large vs. small, fast vs. slow) we focus on detecting more local patterns, i.e. set of diffusion traces whose dynamics can be accurately and compactly described by the same MASD model. This can be accomplished by grouping traces into a set of H clusters, where each cluster C_h can be accurately described by a MASD model governed by parameters Θ_h. Assuming that the number of clusters H is known, we can apply this simple procedure that alternates clustering of traces and parameter estimation.

1) *Initialization*: produce a random partition of traces in $\mathbb{D}$ in H clusters $\{C_1, \cdots, C_H\}$ and determine the parameters Θ_h of the MASD model that maximize the likelihood over traces in C_h.
2) *Update clusters*: At each iteration t, compute
$$h_i^{(t)} = \underset{h=\{1,\cdots,H\}}{\arg\max} \log P(\mathbb{D}_i|\Theta_h),$$
for each trace $\mathbb{D}_i$. If $h_i^{(t)} \neq h_i^{(t-1)}$ then swap $\mathbb{D}_i$ from $C_{h_i^{(t-1)}}$ to $C_{h_i^{(t)}}$.
3) *Update models*: For each $1 \leq h \leq H$, compute
$$\Theta_h = \underset{\Theta}{\arg\max} \prod_{\mathbb{D}_i \in C_h} P(\mathbb{D}_i|\Theta),$$
by applying the EM algorithm.
4) *Convergence*: If the percentage of swaps observed in the current iteration is below a threshold φ, than output the current partiton $\{C_1, \cdots, C_H\}$, otherwise go to step (2).

This is an instance of prototype-based clustering, where we employ the likelihood $P(\mathbb{D}_i|\Theta_h)$ to measure how-well the set of parameters Θ_h represents the dynamics observed in each considered trace $\mathbb{D}_i$. We shall evaluate empirically the effectiveness of such procedure and its scalability in Section V-B.

The clustering procedure may detect groups of diffusion traces whose dynamics exhibit different levels of complexity which, in the HMM framework, translates naturally into the number of hidden states. To automatically tune the number of states that are best suited to describe the underlying patterns in each cluster C_h we need a criterion that realizes a trade-off between quality of fit and complexity of model. For this purpose, we resort to the *Bayesian Information Criterion (BIC)* [14]. Given a cluster C_h and a set of candidate models having different degree of complexity, i.e. number of states, we select Θ_h^* as:

$$\Theta_h^* = \underset{\Theta}{\arg\min} -2\ \mathcal{L}(C_h|\Theta) + k_\Theta \cdot log(\sum_{\mathbb{D}_i \in C_h} |\mathbb{D}_i|)$$

where:

- $\mathcal{L}(C_h|\Theta) = \prod_{\mathbb{D}_i \in C_h} P(\mathbb{D}_i|\Theta_h)$,
- k_Θ is the number of independent parameters to be estimated, i.e., the number of state transition probabilities in left-to-right schema, plus the number of emission probabilities in all the K states, plus one λ in each state, $k_\Theta = \sum_{k=1}^{K} k + K(M-1) + K = K(K+1)/2 + KM$.

V. Experimental evaluation

In this section, we assess our proposed MASD model by mean of various experimental analyses on both synthetic and real-world adoption logs. More in details, our analysis will cover the following aspects:

1) In synthetic data with planted clustering structure, we assess the quality and accuracy of the learning framework in "reconstructing" the known clustering.
2) We also assess convergence and stability of the learning framework, as well as its efficiency on synthetics data.
3) On real-world traces we investigate whether the framework is able to detect clearly distinguishable clusters, w.r.t. complexity of the models, size and speed of the diffusions, as well as the population of users involved.
4) We also study the propensity of users in adopting an item in a state (on real-world data).
5) Finally, we measure the accuracy of MASD model in two different predictive tasks.

Implementation details. The learning procedure for MASD has been developed in Java by extending the *Jahmm* libraries[2]. The learning procedure stops when we observe less that 1% improvement on log-likelihood. All experiments are run on a Intel Xeon 2.2 Ghz with 6 cores and 16 GB memory.

A. Synthetic data generation

We generate synthetic data with planted clustering structure in two steps. First, we generate a set of MASD models as the seeds of the clusters to be generated. In this step, fixing the number of hidden states, we draw model parameters from a fully random process: the upper triangular transition probability matrix is generated randomly, the user emission probabilities $\phi_{u,j}$ are generated uniformly at random and then normalized, and finally the rates of adoption λ_j are sampled from a Gamma distribution (shape=2, scale=0.3).

In the second step actual traces are generated using the following protocol: (i) sample the trace length from a Poisson distribution with mean 50, (ii) sample uniformly at random a model for the generative process, and finally, (iii) generate adoptions by using the selected MASD model as a generative model.

B. Evaluation on synthetic data

Given the procedure to generate synthetic data with known clustering structure, we apply our learning framework to the synthetic data to see to which extent it accurately "reconstructs" the known clusters. To exhaustively evaluate this clustering and fitting procedure we vary both the number of data partitions ($\{4, 8, 16, 32\}$) as well as the size of training data (between $2.5k$ to $25k$). Each experiment is run 10 times.

In Figure 3 we report the results of the evaluation of this "clustering reconstruction" task in terms of *Rand index.*[3]

[2] code.google.com/p/jahmm/

[3] http://en.wikipedia.org/wiki/Rand_index

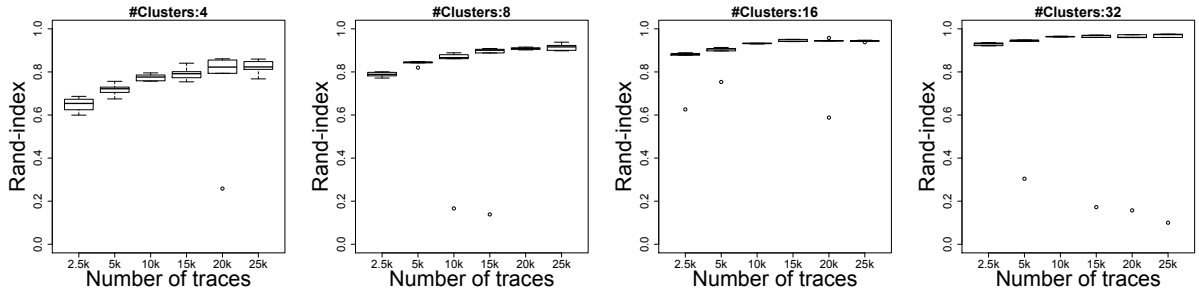


Fig. 3: *Accuracy in the "clustering reconstruction" task on synthetic data with planted clusters, measured in terms of Rand index.*

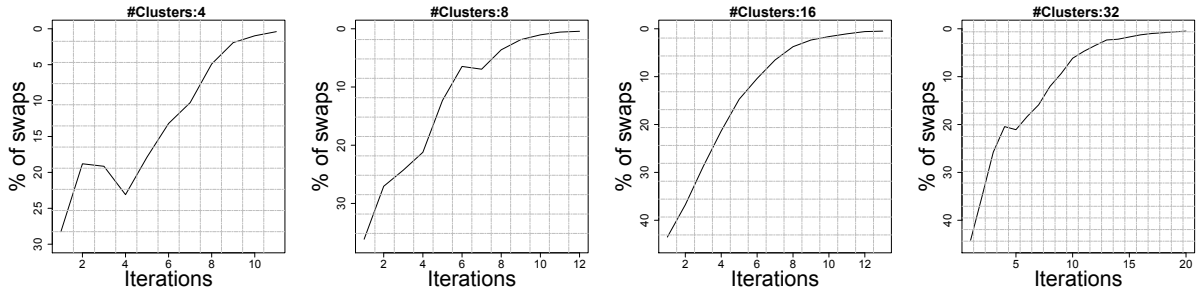


Fig. 4: *Convergence rate of the clustering/learning process, measured as the percentage of swaps observed at each iteration.*

We can see that the values are generally very high. As expected the Rand index grows for larger number of clusters and with the size of observed data. The former is due to the Rand index definition, while the latter is due to the learning procedure, which provides better estimate of parameters as it observes more data.

The convergence rate and the stability of the clustering procedure can be assessed by considering the number of swaps, i.e. the total number of changes in trace-to-model assignments recorded in consecutive iterations. In Figure 4 we report the percentage of swaps at each iteration recorded on the dataset with $10k$ traces. Overall, these empirical results confirm both the effectiveness and stability of the clustering and learning procedure.

	# clusters			
# traces	4	8	16	32
2.5k	109 ± 22	304 ± 50	674 ± 90	$1.5k \pm 125$
5k	247 ± 54	829 ± 250	$2.1k \pm 449$	$4.6k \pm 1.5k$
10k	577 ± 60	$1.2k \pm 279$	$3.6k \pm 501$	$9.8k \pm 1.2k$
15k	1085 ± 245	$1.8k \pm 547$	$4.9k \pm 544$	$11k \pm 2.9k$
20k	$1.2k \pm 417$	$3.7k \pm 931$	$7.2k \pm 1k$	$19k \pm 5k$
25k	$2.2k \pm 659$	$4k \pm 1.5k$	$9.7k \pm 1.2k$	$35k \pm 2.5k$

TABLE II: *Learning time (in secs) on synthetic data.*

Finally, in Table II, we summarize the running times (in seconds) of the overall learning phase. The procedure scales linearly with the size of data and the number of models and takes less than 10 hours to run in the worst among the considered settings.

C. Evaluation on real data.

We next evaluate our model in detecting and characterizing different patterns of adoption on real data. For this purpose, we consider two datasets, namely MovieLens[4] and Yahoo Meme. The first dataset collects explicit user ratings on movies: for us, each movie is a certain item diffused over the network and each adoption corresponds to a user rating a particular movie. The second dataset is a sample of temporal snapshot (from 1 Jan. till 31 Dec. 2010) extracted from Yahoo Meme, a microblogging service[5], in which users can share different kinds of information called "memes". Here a diffusion trace is defined by the set of users who have shared the same meme and their corresponding timestamp of the sharing.

Before studying their diffusion dynamics, we perform some basic preprocessing of the input datasets, disregarding those traces whose length differs from the mean more than two times the standard deviation. A brief summary of the main properties of both the datasets is given in Table III. The main difference between the two datasets is the rate of adoptions, which is clearly higher for Yahoo Meme than MovieLens, as expected given the different nature of the two datasets (memes vs. movies). For the sake of presentation, we choose 5 as the number of clusters to be identified. As explained in Section IV we employ BIC to select the number of states in the prefixed range [2, 12].

Table IV summarizes the main properties of the clusters found by our model. Here, the size of each cluster is the number of traces that are associated to it, and we can observe a mix of small and large size clusters on both datasets. As expected, each cluster reflects the patterns of diffusion with a different level of complexity, which is indicated by the varying number of adoption stages. The length of a trace represents the number of adoptions: while on Yahoo Meme we cannot

[4]Publicly available at http://grouplens.org/datasets/hetrec-2011/

[5]Discontinued in May 25, 2012

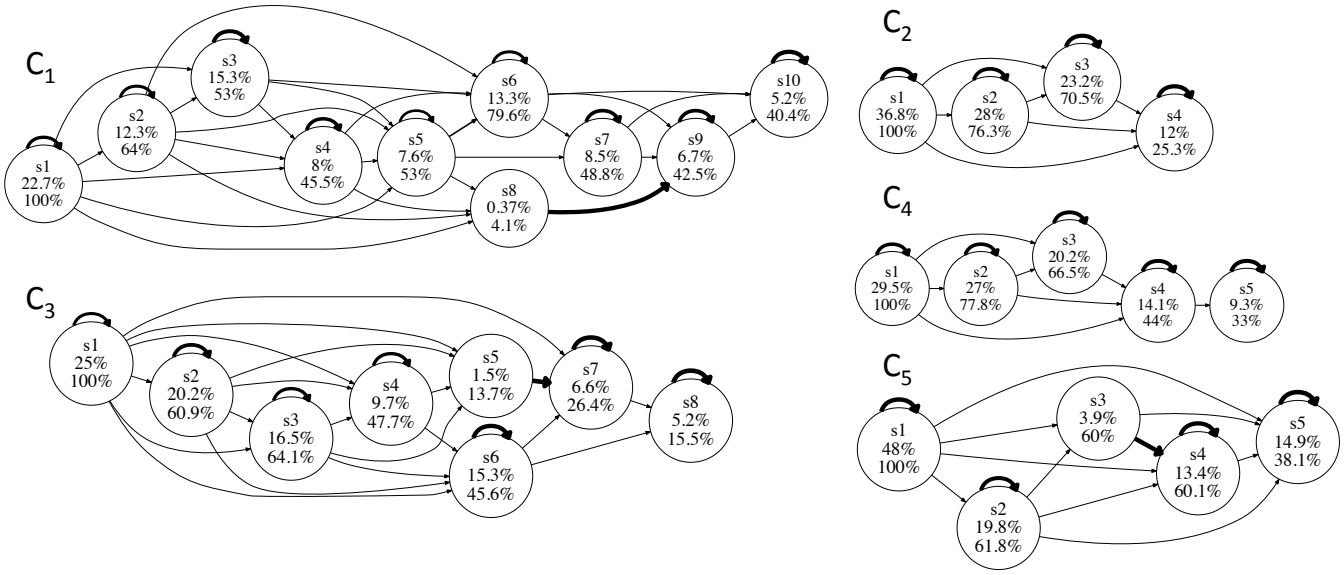


Fig. 5: *Graphical representation of the MASD model produced in Yahoo Meme dataset. The thickness of arcs indicate the strength of transition probability between the states. The numbers inside each state represent: (i) the index of the adoption stage in the left-right model, (ii) the average percentage of adoptions and (iii) the percentage of traces that involve the corresponding stage. These statistics are computed by applying the Viterbi algorithm to find the most likely sequence of hidden states for a diffusion trace.*

	MovieLens	Yahoo Meme
Number of users	2,113	21,236
Diffusion traces	7,780	33,421
Overall adoptions	751,840	954,871
Avg. adoptions per user	356	45
Avg. adopters per trace	97	29
Avg. time span of traces (days)	1912	65
Avg. adoption rate	0.02	6.25

TABLE III: *Datasets statistics.*

	states	*size (traces)*	*avg. trace length*	*avg. time span of traces (days)*	*avg. rate of adoption*
C_1	7	830	33	1843	$2.5E-4$
C_2	7	210	472	2222	$2.0E-4$
C_3	7	3935	37	2373	$1.6E-4$
C_4	8	2438	188	2528	$2.7E-4$
C_5	5	367	55	593	$1.6E-4$

(a) MovieLens

	states	*size (traces)*	*avg. trace length*	*avg. time span of traces (days)*	*avg. rate of adoption*
C_1	10	18749	26	49	$7.7E-3$
C_2	4	2485	40	72	$9.3E-2$
C_3	8	7046	21	55	$8.3E-3$
C_4	5	2971	39	59	$3.0E-2$
C_5	5	2120	50	90	$4.0E-2$

(b) Yahoo Meme

TABLE IV: *Main properties of the diffusion traces within the detected clusters.*

	C_2	C_3	C_4	C_5
C_1	0.12	0.06	0.07	0.15
C_2		0.17	0.03	0.16
C_3			0.09	0.27
C_4				0.16

(a) MovieLens

	C_2	C_3	C_4	C_5
C_1	0.13	0.10	0.10	0.13
C_2		0.08	0.07	0.07
C_3			0.08	0.09
C_4				0.07

(b) Yahoo Meme

TABLE V: *JSD between cluster-specific multinomial distributions over users.*

appreciate a significant difference among clusters with respect to this dimension, the resulting partitioning on MovieLens clearly differentiates between popular and less popular traces.

The evolution of diffusion patterns for each cluster can be nicely described by plotting the corresponding model. Figure 5 graphically shows the stages of diffusion in all the clusters of Yahoo Meme.

We also discuss the evolution of diffusion rates (λ_j) in different states of the model. These are reported in log-scale in Figure 6. In both the datasets, the rates are higher in the initial state, i.e., the expected time delay between consecutive adoptions is low. The rate quickly drops from the first state and we observe different trends from this point onwards. In MovieLens, diffusion rates start increasing towards the later states, where as, in Yahoo Meme we observe an entirely opposite trend as the rate falls mostly sharply towards the later states. The evident differences among the evolution of diffusion rates in different clusters empirically confirms the need of detecting groups of traces that share the same diffusion patterns, rather than having a single representative model of the entire dataset.

In order to assess user involvements in different clusters, we compute $P(u|C_h)$ as the probability of observing a user adoption u in cluster C_h. The pairwise comparison between these distributions provides us an insight on how the considered clusters tend to interest different user segments of the entire population. Table V reports the Jenson-Shannon divergence (JSD) (values are bounded in $[0, 1]$ due to the use of the base 2 logarithm for the computation) between cluster-specific multinomial distributions over users. This analysis suggests that while the difference between some clusters can

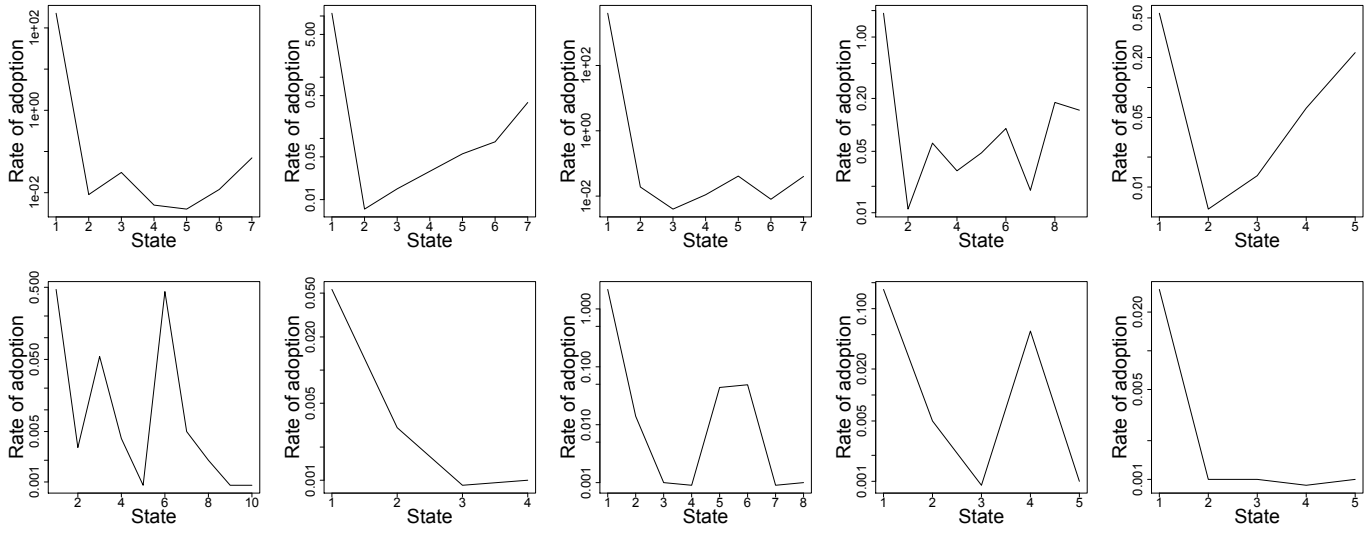


Fig. 6: *Different diffusion rates* (log-*scale) in the five clusters in* MovieLens *(top) and* Yahoo Meme *(bottom).*

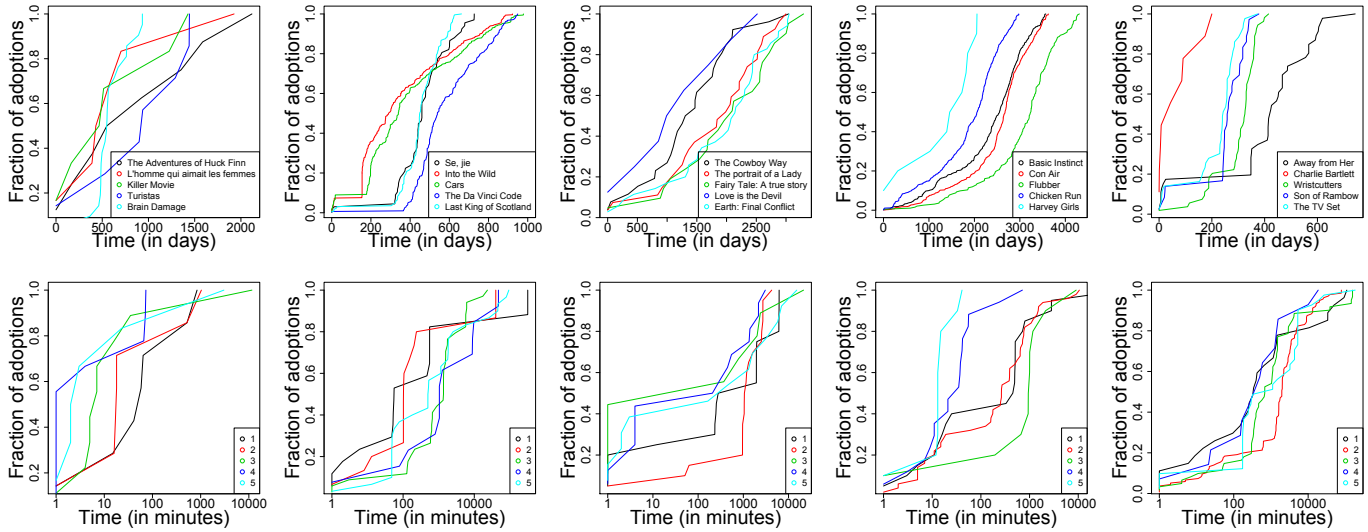


Fig. 7: *Adoption patterns, in the 5 clusters,* MovieLens *(top) and* Yahoo Meme *(bottom).*

be explained at user level, other clusters tend to involve the same users but with different temporal dynamics.

We also look at the differences among the diffusion patterns of different clusters at the level of individual traces. For this, first, we select a few representative traces in each cluster. These traces are the top-5 w.r.t. minimizing the perplexity for the considered model. Formally:

$$perplexity(\mathbb{D}_i|\Theta_h) = exp\left\{-\frac{\log P(\mathbb{D}_i|\Theta_h)}{|\mathbb{D}_i|}\right\}, \qquad (4)$$

where, the numerator is the log likelihood of the adoption trace given the model and denominator is the length of the trace (number of adoptions). Then, for each cluster we plot in Figure 7 the fraction of adoptions with respect to time for the top-5 representative traces. These plots show homogeneous patterns of diffusions for the traces belonging to the same cluster; for instance, the second cluster in MovieLens contains movies that are diffused slowly in the beginning, however, their popularity increases sharply after a certain point.

Finally, we study to how many stages a user is associated. We compute user-specific distributions in each stage by exploiting the output of the Viterbi algorithm. We define $P(\mathcal{S}(u) = j)$ as the probability that for a given user, an adoption happens in the j-th stage of the diffusion process, and we associate the user to a stage only if $P(\mathcal{S}(u) = j) \geq 0.1$. In Figure 9 we plot the distribution of the number of stages that each user is associated to. On both the datasets, the figures suggest that the majority of the users are associated to 1, or maximum 2, stages of diffusion.

Furthermore, we compute for each user u the expected stage in which we observe his adoption, as:

$$E\left[\mathcal{S}(u)\right] = \sum_{j=1}^{K} j \cdot P(\mathcal{S}(u) = j).$$

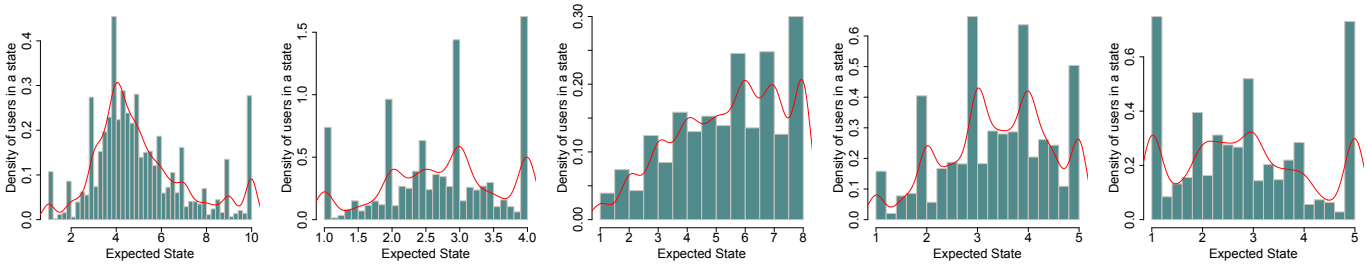


Fig. 8: *Distribution of expected stage of adoption for users in Yahoo Meme.*

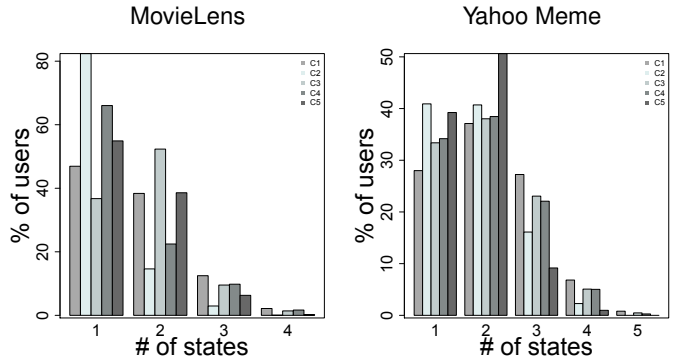


Fig. 9: *Distribution of the number of stages associated to each user.*

We plot the densities of the expected stage of adoption for each user on Yahoo Meme data in Figure 8. We can observe again that different clusters exhibit very diverse behavior. An interesting observation is that a very large portion of the users have an integer value for the expected stage of adoption: this is because they have probability 1 of activating in one stage and 0 in all the other stages.

D. Predictive tasks

The joint modeling of users' behavior and temporal dynamics allows the application of the proposed model to interesting prediction scenarios. Assume that we have trained our models on observed data. For a new item that is only partially observed (up-to time t), we consider the following two prediction questions:

- *Q1*: Which users will adopt the item within a considered time window (e.g 1 week)? This information, if accurate, enables dynamic marketing strategies where the focus is to target specific user segments that are likely to adopt the item in the near future.
- *Q2*: How many users will adopt the item in a considered time window? This determines the future popularity of an item and an accurate estimate of this could tune the effectiveness of advertisement.

In order to perform these evaluations we randomly split diffusion traces in $\mathbb{D}_{train}$ and $\mathbb{D}_{test}$, where the latter accounts for 20% of the overall available data. We use $\mathbb{D}_{train}$ for learning model parameters and $\mathbb{D}_{test}$ to evaluate model predictions. The first step of the prediction tasks is to select a portion of each trace $\mathbb{D}_i \in \mathbb{D}_{test}$ (that represents a partially observed trace) and associate it to a model that better describes its dynamics, or, the one that maximizes its log-likelihood. Note that the quality of the fitting is expected to change when we increase the length of observed traces in the evaluation set. To account for this phenomenon we evaluate the prediction accuracy on 3 settings by varying the lengths of the traces among 60%, 50% and 40%. The accuracy of the prediction is measured on the remaining part of the evaluation traces.

We address our first prediction question by means of a simulation-based method. Here, given a trace $\mathbb{D}_i$ observed until time t_a, we find its most likely current state of diffusion after determining the model that suits its dynamics the most. The current diffusion state is easy to find by applying Viterbi (given the representative MASD model). From the current state, we generate multiple samples according to the generative process. Assuming that we are interested in predicting user adoptions within the time interval $[t_a, t_b]$, we stop the generative process as soon we sample an adoption at a time greater or equal to t_b. We repeat this process for a fixed number of runs ($1k$ in our setting) and compute the probability of observing user adoptions in the simulated traces. As baseline competitor, we rely upon a k-nearest neighbors algorithm. For this, first, we compute the Jaccard index between the set of users in the partially observed evaluation trace and each other adoption trace in $\mathbb{D}_{train}$. Then, we select the top-k most similar traces and aggregate their information within the considered prediction window to compute the probabilities of users' adoptions.

Table VI provides the area-under-the-curve (AUC) values recorded on this evaluation, considering three prediction time windows, up to 30 days for MovieLens and 24 hours for Yahoo Meme. Our model consistently outperforms the k-NN baselines, for all considered values of k. As expected, the accuracy of our method increases with the length of the evaluation trace used for fitting, but MASD is still able to achieve satisfactory results on the most difficult among considered settings.

We address the second prediction question by considering similar techniques that we used for the previous question. Recall that our focus now is to predict the number of adoptions in a given time window. We generate the predicted number of adoptions for a partially observed trace by averaging the length of the synthetic data sampled from the model. The baseline competitor computes the weighted average (based on the Jaccard index) of the lengths of the k-nearest neighbors. The values of mean absolute error (MAE) between the actual and estimated number of adoptions on both dataset are shown in Table VII. Again, MASD outperforms the k-NN baseline in all considered settings.

	60% partial observation				50% partial observation				40% partial observation			
Time Window	MASD	k-NN (60, 80, 100)			MASD	k-NN (60, 80, 100)			MASD	k-NN (60, 80, 100)		
30 days	**0.70**	0.54	0.55	0.55	**0.69**	0.55	0.55	0.55	**0.69**	0.54	0.54	0.55
21 days	**0.69**	0.55	0.55	0.55	**0.69**	0.54	0.54	0.55	**0.68**	0.54	0.54	0.54
14 days	**0.69**	0.54	0.54	0.54	**0.69**	0.54	0.54	0.55	**0.69**	0.53	0.54	0.54

(a) Movielens

	60% partial observation				50% partial observation				40% partial observation			
Time Window	MASD	k-NN (60, 80, 100)			MASD	k-NN (60, 80, 100)			MASD	k-NN (60, 80, 100)		
60 min.	**0.83**	0.73	0.74	0.75	**0.82**	0.72	0.73	0.74	**0.82**	0.72	0.74	0.74
30 min.	**0.82**	0.72	0.73	0.74	**0.81**	0.71	0.72	0.72	**0.81**	0.71	0.72	0.73
15 min.	**0.81**	0.68	0.70	0.70	**0.80**	0.69	0.70	0.71	**0.81**	0.66	0.68	0.69

(b) Yahoo Meme

TABLE VI: *Area under the curve (AUC) for predicting single user activations in different time windows. The baseline procedure is evaluated for three selections of k and three different splits of propagations.*

	60% partial observation				50% partial observation				40% partial observation			
Time Window	MASD	k-NN (60, 80, 100)			MASD	k-NN (60, 80, 100)			MASD	k-NN (60, 80, 100)		
30 days	**3.42**	3.90	3.90	3.92	**3.71**	3.90	3.91	3.92	**4.61**	5.14	5.17	5.17
21 days	**2.61**	3.02	3.02	3.03	**2.88**	3.02	3.03	3.03	**3.61**	3.96	3.97	3.98
14 days	**1.93**	2.22	2.23	2.23	**2.16**	2.23	2.23	2.23	**2.69**	2.90	2.91	2.91

(a) Movielens

	60% partial observation				50% partial observation				40% partial observation			
Time Window	MASD	k-NN (60, 80, 100)			MASD	k-NN (60, 80, 100)			MASD	k-NN (60, 80, 100)		
60 min.	**3.57**	5.56	5.61	5.63	**5.32**	7.20	7.24	7.25	**7.46**	9.01	9.08	9.09
30 min.	**3.01**	4.65	4.69	4.71	**4.66**	6.13	6.17	6.15	**6.69**	7.82	7.89	7.93
15 min.	**2.49**	3.23	3.25	3.26	**4.53**	5.89	5.92	5.93	**5.50**	5.63	5.66	5.68

(b) Yahoo Meme

TABLE VII: *Mean absolute error (MAE) for predicting aggregate user activations in different time windows. The baseline procedure is evaluated for three selections of k and three different splits of propagations.*

VI. Conclusions

In this paper we introduce MASD, a stochastic framework for modeling users' adoptions and the different stages of diffusion of innovations. In continuity with Roger's theory, our model focuses on the two main dimensions that can explain the spread of new items through a population, namely the users and their propensity to adopt a new item in a specific stage of diffusion, and the speed at which adoptions happen in the various stages. To capture the evolution dynamics between stages of adoptions MASD relies on a *left-to-right* hidden Markov model. The proposed learning procedure allows us to detect fine-grained patterns in the underlying diffusion process. The experimental evaluation over real-world data confirms the accuracy of the learning framework and its ability to detect distinct, and interesting, patterns of adoption.

For future work, we plan to investigate an extension of the proposed model to account for social influence dynamics. In this scenario, the likelihood of adoption for each user is naturally expected to increase as more of his social peers adopt the item. Moreover, each stage of adoption could be further characterized in terms of virality, hence enabling the detection, characterization and prediction of stages in which the adoption of a product will become viral.

Repeatability. All software (sources and executables) and the sample from Movielens used in our experiments are available at https://github.com/yasirm/ICDM2014.

Acknowledgments. This work was partially supported by MULTISENSOR project, funded by the European Commission, under the contract number FP7-610411.

References

[1] R. Bakis. Continuous speech recognition via centisecond acoustic states. *Acoustical Society of America Journal*, 59:97, 1976.
[2] E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic. The role of social networks in information diffusion. In *WWW*, 2012.
[3] C. M. Bishop et al. *Pattern recognition and machine learning*, volume 1. springer New York, 2006.
[4] C. Budak, D. Agrawal, and A. El Abbadi. Diffusion of information in social networks: Is it all local? In *ICDM*, 2012.
[5] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39:1–38, 1977.
[6] S. Goorha and L. Ungar. Discovery of significant emerging trends. In *KDD*, 2010.
[7] J. Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397, 2003.
[8] J. Leskovec, L. Backstrom, and J. Kleinberg. Meme-tracking and the dynamics of the news cycle. In *KDD*, 2009.
[9] M. Mathioudakis and N. Koudas. Twittermonitor: trend detection over the twitter stream. In *SIGMOD*, 2010.
[10] I. Mele, F. Bonchi, and A. Gionis. The early-adopter graph and its application to web-page recommendation. In *CIKM*, 2012.
[11] E. M. Rogers. *Diffusion of innovations*. Free Press, 5th edition, 2003.
[12] D. M. Romero, B. Meeder, and J. Kleinberg. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter. In *WWW*, 2011.
[13] D. Saez-Trumper, G. Comarela, V. Almeida, R. Baeza-Yates, and F. Benevenuto. Finding trendsetters in information networks. In *KDD*, 2012.
[14] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6:461–464, 1978.
[15] J. Yang and J. Leskovec. Patterns of temporal variation in online media. In *WSDM*, 2011.