9-2015

Latent Factors Meet Homophily in Diffusion Modelling

Duc Minh LUU

Ee-peng LIM
Singapore Management University, eplim@smu.edu.sg

DOI: https://doi.org/10.1007/978-3-319-23525-7_43

Latent Factors Meet Homophily in Diffusion Modelling

Minh-Duc Luu and Ee-Peng Lim

School of Information Systems, Singapore Management University,
80 Stamford Road, Singapore 178902.
mdluu.2011@smu.edu.sg, eplim@smu.edu.sg

Abstract. Diffusion is an important dynamics that helps spreading information within an online social network. While there are already numerous models for single item diffusion, few have studied diffusion of multiple items, especially when items can *interact* with one another due to their inter-similarity. Moreover, the well-known homophily effect is rarely considered explicitly in the existing diffusion models. This work therefore fills this gap by proposing a novel model called *Topic level Interaction Homophily Aware Diffusion* (TIHAD) to include both latent factor level interaction among items and homophily factor in diffusion. The model determines item interaction based on latent factors and edge strengths based on homophily factor in the computation of social influence. An algorithm for training TIHAD model is also proposed. Our experiments on synthetic and real datasets show that: (a) homophily increases diffusion significantly, and (b) item interaction at topic level boosts diffusion among similar items. A case study on hashtag diffusion in Twitter also shows that TIHAD outperforms the baseline model in the hashtag adoption prediction task.

1 Introduction

Ubiquitous presence of online social networks (OSN) has made information diffusion an important topic that attracts much research interests. While many items may diffuse in a social network simultaneously, most existing models of diffusion are built upon *independent* contagion assumption whereby the diffusion of each item is assumed (at least implicitly) to happen independent of other items. The interaction among items during diffusion is thus left out of the picture. This is obviously not true in the complex dynamics of diffusion process. For instance, the diffusion of iPhones in the Facebook friendship network may interact favorably with that of iPad; and the diffusion of a catchy phrase on Twitter also aids the diffusion of its variants.

Interaction among items. Modeling these interactions is crucial in both theory and practice since it helps us understand the detailed dynamics of multiple item diffusion. It is also valuable for business to develop suitable strategies to promote diffusion of their own items considering the other items that have been diffused recently or are being diffused. It may be good to time the diffusion of a new item with the diffusion of other similar items (possibly by the business or other businesses) to achieve a larger reach. This idea of diffusion with item interaction can be further illustrated in the following motivating example.

Example. *A user may be inspired to watch the movie version of "Hunger Games" after observing some neighbors already read the book. Moreover, if both the book and the movie versions were adopted by a neighbor, the user will even be more likely to adopt the movie than if only one of them was adopted by the neighbor (as he may be more convinced that the movie is good in the former case).*

The example not only highlights that diffusion of an item can support that of another *similar* item but suggests other deeper ideas which will distinguish our work from the rest. These ideas are:

1. *The more similar items are, the more interaction will happen between them in diffusion*. In other words, item similarity can be used as a proxy for *item interaction*. This idea will be formulated in Section 3.2 where we propose a general diffusion framework for modeling item interaction when there is more than one item diffusing.
2. *Whether or not a user adopts an item i is affected not only by neighbors who adopted exactly the item but also by those who adopted other items*. A neighbor who already adopted another item i' can still influence the decision (see Example) as i' may be very similar to i.
3. *Each neighbor's social influence on a user's adoption decision should include all contributions from a set of items adopted by the neighbors*, not just limited to one item as in the existing models.

Homophily Factor. Another important aspect which also has great impact on item diffusion, is the well-known *homophily* phenomenon. Homophily refers to the tendency of individuals to associate and bond with similar others. It is well known that homophily affects the mechanisms in which item diffusion happens, be it innovation [14], information [3] or behavior [2]. Thus, it is important to integrate homophily into diffusion models so that we can better quantify its effect on diffusion. In this work, we assume a global homophily level of the network and learn it from the diffusion cascade data. Given that networks with homophily involves more similar users connecting with one another, it also plays a role in determining if an item can more smoothly diffuse to across the network links.

Research Objectives. In this paper, we therefore propose to consider the above two factors in the design of a new diffusion model. To involve both item similarity (which helps to estimate item interaction) and user similarity (due to homophily), our modeling approach employs latent factors (LF) to represent both items and users (e.g. [13], [10]) where each user or item is represented as a vector in a common feature space with dimension much smaller than that of items and users. The similarity between two items (or users) can then be defined by the cosine similarity of the respective item (or user) vectors. Unlike the collaborative filtering approach taken by recommender systems, our diffusion modeling work also consider social influence among users. Although there are recently hybrid models ([9], [12]) which combine latent factor approach with social networks, they still do not model a user adopting an item influenced by the neighbors' past adoption of similar items and the strength of relationships with these neighbors. Based on our proposed model, we seek to answer some interesting research questions related to multiple-item diffusion in homophily networks.

Summary of contributions. In summary, our work makes the following contributions.

- We develop an extended diffusion framework which incorporates both item interaction and homophily into modeling diffusion. To the best of our knowledge, this is the first attempt to combine the two factors. The framework is flexible and can offer useful insights to multiple item diffusion.
- We propose a specific diffusion model based upon the new framework. This model, known as TIHAD, utilizes latent factors to capture item interaction and homophily effect for effective modeling diffusion processes of multiple items.
- We formulate the parameter learning of model as a constrained optimization problem, and devise an effective learning algorithm using Projected Gradient Descent.
- We conduct experiments on both synthetic and real datasets to show that: (a) homophily increases diffusion significantly, and (b) item interaction at topic level boosts diffusion among similar items. We also shows that TIHAD outperforms the baseline model in the hashtag adoption prediction task.

Paper Outline. We will next give an overview of the related works. In Section 3, we present our proposed diffusion model known as TIHAD. The learning of this model is given in Section 4. Section 5 describes experiments that evaluate the TIHAD using both synthetic and real datasets. We finally conclude the paper in Section 6.

2 Related works

Our work is closely related to very well studied adoption and diffusion modeling research: (i) Latent Factor models and (ii) Social Influence models. In the following sections, we briefly review these research works and relate them with our work.

2.1 Latent Factor Models

These models ([16], [13], [10]) take a user-item adoption matrix and factorize it into a set of user and item vectors with f dimensions where f is much smaller than the number of users or items. For each item i, a latent factor vector $\boldsymbol{q}_i \in \mathbb{R}^f$ is derived and it contains the relevance weights of the latent factors for the item i. Similarly, a latent factor vector $\boldsymbol{p}_u \in \mathbb{R}^f$ is derived for each user u to represent the weights u has for the latent factors. Thus, the amount of interest u has towards item i can be defined as the inner product $\boldsymbol{p}_u^T \boldsymbol{q}_i$. Unlike latent factor models which focus on user-item interactions only, our work considers both user-item and item-item interactions in the diffusion setting. We are therefore also interested in the effect of item similarity. We exploit the latent factor space by defining the similarity between two items i and j as the inner product $\boldsymbol{q}_i^T \boldsymbol{q}_j$.

For better interpretability, many Latent Factor (LF) models (see [13], [15]) require latent factor vectors to have positive elements. We also follow this practice and consider only positive latent factor vectors. Although LF models enjoy the benefit of dimension reduction by matrix factorization, they do not consider the underlying social network which forms the substrate over which diffusion occurs. To address this shortcoming, recent research proposed to exploit social influence in the modeling of user-item adoptions (or ratings).

2.2 Social Influence and Diffusion Models

Social influence modeling works takes into account social interest and social trust as additional input to achieve better accuracy for recommendation ([9], [12], [18], [17], [4]). These works proposed various ways of modeling the social dimension such as factorizing the social network graph ([12]) or modeling social factors of users as another set of latent factors ([17], [4]). While these works focus on recommendation tasks, they are similar to diffusion models in that both estimate social influence on user-item adoptions. Social diffusion models on the other hand consider only influence from a subset of neighbors, called the set of *active* neighbors $\mathcal{A}_u$, who adopt exactly the target item ([6], [8], [11]). For example, Linear Threshold (LT) model is a social diffusion model which estimates social influence by the sum of weights of active neighbors. Thus, its standard form is

$$social\ influence = \sum_{v \in \mathcal{A}_u} w_{v,u} \tag{1}$$

As pointed out in our motivating example, items similar to the item being diffused i can affect diffusion. Even though a neighbor has not yet adopted item i, he can still affect the target user's decision on adopting i, when the neighbor adopted item(s) similar to i. Such a diffusion scenario has been largely overlooked in the existing social diffusion models.

3 Proposed Framework and Model

Before we present our proposed modeling framework and the TIHAD model, we first introduce the notations used in the problem formulation.

3.1 Basic Notations

We represent a social networks as a (directed), weighted graph $G = (U, E)$ whose nodes represent users and edges represent links among the users. For each edge (u, v), the edge weight $w_{v,u}$ represents the social influence that v exerts on u. To model diffusion over the network during a time period, we bin the continuous time into discrete time steps $\{1, 2, \cdots, T\}$ and consider adoptions in each step.

Denote adoption decision of a user u on item i at time step t as $a_{u,i,t}$. At first sight, it seems that $a_{u,i,t}$ is simply a binary label which is 1 when u adopt i and 0 otherwise. However, it is often that a user does not adopt an item because he has not been exposed to the item. It is thus incorrect to assume that he rejects the item, and underestimate his preference for the item. We can avoid this by considering, at each time step, only items which are exposed to the user. When the user did not adopt an item he has exposed to, we say that the case is a non-adoption. We call these user-exposed items as the candidate items in Definition 1, which in turn help us to define adoption labels properly in Definition 2.

Definition 1 (Candidate item)**.** *At a given time step t, a candidate item for a user is an item that: (i) he has not yet adopted before t; and (ii) he is exposed to it through some source (e.g., recent adoptions by his neighbors). The set of candidate items for a user u at time t is denoted by $C_{u,t}$.*

Definition 2 (Adoption label)**.** *Given an item $i \in C_{u,t}$, adoption label $a_{u,i,t}$ is a binary variable which is 1 if u adopts i at time t and 0 otherwise.*

3.2 Framework

Our proposed framework extends the latent factor model framework by considering both personal interest and social influence in the modeling of user-item adoption at different time steps. Personal interest is estimated by user-item similarity in a latent space and social influence is an aggregation of individual influences from neighbors. However, that influence from a neighbor $v \in N_u$ (the set of neighbors of u) now depends on: (i) the link weight $w_{v,u}$, and (ii) the interaction level between item i and a certain set of items adopted by v. We also follow common practice (e.g. [9]) by including in the framework global bias μ, user bias b_u and item bias b_i.

For easy reading, we first state the core formula of the framework in Eqn. (2) and provide the reasoning behind the formulae subsequently. By denoting personal interest and social influence as $\phi(u,i)$ and $\sigma(u,i,t)$ respectively, we can express the framework as follows (the logic behind will be explained soon).

$$\widehat{a}_{u,i,t} \coloneqq \mu + b_u + b_i + \overbrace{\boldsymbol{p}_u^T \boldsymbol{q}_i}^{\phi(u,i)} + \overbrace{\sum_{v \in N_u} w_{v,u} \cdot \lambda\left(v,t,i\right)}^{\sigma(u,i,t)} \tag{2}$$

where we introduce the following

1. $w_{v,u}$: link weight, which will later be estimated by a function of user similarity parameterized by the so-called *homophily level*, which will be denoted as h
2. $\lambda\left(v,t,i\right)$: the *interaction level* (will be defined formally later) between the items adopted by v and item i at time step t.

Our framework adapts the general formula by proposing in Eqn. (2) a novel estimation of social influence term $\sigma(u,i,t)$ and a homophily derived link weight $w_{v,u}$. As can be seen from the definitions, the estimation will incorporate both *item interaction* and *homophily factor*. To keep the framework tractable, we assume the latent factors are static. Given this framework, we can now apply it for modeling *interacting diffusion* processes of items over a social network as follows.

Framework (Interacting Diffusion of Items)**.** *Consider a set of items I and a social network G. For each such candidate item i, its adoption label $\widehat{a}_{u,i,t}$ can be estimated by Eqn.* (2)*. Candidate i will be adopted by u if the estimation is close enough to 1 (i.e., $\widehat{a}_{u,i,t} \geq 1 - \theta$). Thus, at each time step, a user can adopt several candidate items which satisfy this criterion. The process continues until no more adoption can happen.*

We proceed by providing the logic behind Eqn. (2) of our framework. The logic includes two parts: how to define item interaction and how to incorporate homophily.

Item interaction The interaction level depends on a certain set of v's adopted items which can actually affect u's decision. This leads us to the concept of *effective item set* defined as follows.

Definition 3 (Effective item set)**.** *For a given neighbor v of user u, the set of items adopted by v which can influence adoption decision $a_{u,i,t}$ is called* effective item set *from the neighbor at time step t and denoted as $I_{eff}(v,t)$.*

Given effective item set $I_{eff}(v,t)$, we now need to estimate the interaction level $\lambda(v,t,i)$ between the adopted items of v and candidate item i and time step t. We now provide a general estimation of $\lambda(v,t,i)$ in Definition 4.

Definition 4 (Interaction level)**.** *The interaction level $\lambda(v,t,i)$ is defined as the sum of interactions (i.e. similarities) between the effective item set of v and i at time step t.*

$$\lambda(v,t,i) \coloneqq \sum_{j \in I_{eff}(v,t)} \boldsymbol{q}_j^T \boldsymbol{q}_i \tag{3}$$

The social influence from neighbor v will then be $w_{v,u} \times \lambda(v,t,i)$. In total, social influence on u will be estimated by

$$\sigma(u,i,t) \coloneqq \sum_{v \in N_u} w_{v,u} \times \lambda(v,t,i) = \left(\sum_{v \in N_u} \sum_{j \in I_{eff}(v,t)} w_{v,u} \boldsymbol{q}_j \right)^T \boldsymbol{q}_i \tag{4}$$

Note that for directed networks, N_u will be replaced by the followee set of u.

Replace (4) into (2), we obtain our novel estimation for adoption label

$$\widehat{a}_{u,i,t} \coloneqq \mu + b_u + b_i + \boldsymbol{p}_u^T \boldsymbol{q}_i + \left(\sum_{v \in N_u} \sum_{j \in I_{eff}(v,t)} w_{v,u} \boldsymbol{q}_j \right)^T \boldsymbol{q}_i \tag{5}$$

This new estimation allows our framework to capture item interaction. Thus, in the context of interacting diffusion, we expect it to provide a better model than existing models (e.g. [11]). This will be realized later in our experiments on synthetic data.

Incorporating homophily Eqn. (5) involves link weight $w_{v,u}$ which is determined by homophily factor. Due to homophily effect, more similar individuals tend to be connected. We therefore propose to estimate $w_{v,u}$ as an *increasing* function of the similarity between u and v. In other words, for a social network with an underlying homophily level $h \in [0,1]$ (smaller h implies low homophily), we propose to define $w_{v,u}$ as:

$$w_{v,u} \coloneqq g(\boldsymbol{p}_u^T \boldsymbol{p}_v | h) \tag{6}$$

where $g(.)$ is an increasing function parameterized by h. Since weights are in $[0,1]$, we also choose functions g with range in $[0,1]$.

Finally, by replacing Eqn. (6) in (5) and using estimation of $\lambda(v,t,i)$, we obtain Eqn. (7), the main estimation of our framework.

$$\widehat{a}_{u,i,t} \coloneqq \mu + b_u + b_i + \boldsymbol{p}_u^T \boldsymbol{q}_i + \sum_{v \in N_u} g(\boldsymbol{p}_u^T \boldsymbol{p}_v | h) \cdot \sum_{j \in I_{eff}(v,t)} \boldsymbol{q}_j^T \boldsymbol{q}_i \tag{7}$$

3.3 Topic Interaction and Homophily Aware Diffusion (TIHAD) Model

To apply our general framework, we need to give specific definitions for $g(\boldsymbol{p}_u^T \boldsymbol{p}_v|h)$, and $I_{eff}(v,t)$. This leads to our proposed Topic Interaction and Homophily Aware Diffusion (TIHAD) Model.

In TIHAD, we define the function $g(.)$ as a linear function of user similarity $\boldsymbol{p}_u^T \boldsymbol{p}_v$ as follow.

$$g(\boldsymbol{p}_u^T \boldsymbol{p}_v|h) \coloneqq h \cdot (\boldsymbol{p}_u^T \boldsymbol{p}_v),\ \forall(u,v),\ v \in N_u \tag{8}$$

There are other interesting forms of function $g(.)$ including $\left(\boldsymbol{p}_u^T \boldsymbol{p}_v\right)^h$. In this work, we focus on the linear form due to its tractability and leave other forms for future research.

For $I_{eff}(v,t)$, we choose the set of items *adopted recently* by neighbor v. This is based on the common intuition that a user usually pays attention only to those recent items (e.g., Twitter users only focus on recent hashtags from their followees [19]). Thus, for each time t, we choose the effective set as the set of k items which neighbor v adopted most recently with respect to time step t, which we denote as $r_v^{k,t}$. Hence, $I_{eff}(v,t) = r_v^{k,t}$.

The TIHAD model is therefore expressed as Eqn. (9).

$$\widehat{a}_{u,i,t}^{\mathbf{tihad}} = \mu + b_u + b_i + \boldsymbol{p}_u^T \boldsymbol{q}_i + h \boldsymbol{p}_u^T \left[\boldsymbol{S}_t(u)\right] \boldsymbol{q}_i \tag{9}$$

where the matrix $\boldsymbol{S}_t(u)$ is

$$\boldsymbol{S}_t(u) \coloneqq \sum_{v \in N_u} \boldsymbol{p}_v \left(\sum_{j \in r_v^{k,t}} \boldsymbol{q}_j \right)^T \tag{10}$$

$\boldsymbol{S}_t(u)$ can be interpreted as the matrix characterizing the social influence from u's neighbors recent adoption events.

3.4 Linear Threshold with Latent Factors (LTLF)

In the special case when $I_{eff}(v,t) = \{i\}$, Eqn. (4) becomes

$$\tilde{\sigma}(u,i,t) = \left(\sum_{v \in \mathcal{A}_{u,t}^i} w_{v,u} \right) \|\boldsymbol{q}_i\|^2$$

where v now is not an arbitrary neighbor of u but instead an *active* neighbor i.e. one who actually adopted i at time t. This estimation of social influence is obviously an extension of Eqn. (1) commonly used in Linear Threshold models ([6], [1]). Thus, by substituting it into Eqn. (5), we obtain the following model, called Linear Threshold with Latent Factors (LTLF)

$$\widehat{a}_{u,i,t}^{\mathbf{ltlf}} \coloneqq \mu + b_u + b_i + \boldsymbol{p}_u^T \boldsymbol{q}_i + \left(\sum_{v \in \mathcal{A}_{u,t}^i} w_{v,u} \right) \|\boldsymbol{q}_i\|^2 \tag{11}$$

4 Learning of TIHAD Model

We formulate the learning of TIHAD model parameters as a constrained optimization problem, which can be solved by Projected Gradient Descent (PGD). We provide the detailed formula to solve the problem and a pseudocode for model learning. For brevity, we use $\boldsymbol{P}$ and $\boldsymbol{Q}$ matrices to denote user and item latent factors respectively. All parameters of TIHAD then can be compactly represented by $\boldsymbol{\Pi} = (h, \mu, \{b_u\}_{u \in U}, \{b_i\}_{i \in I}, \boldsymbol{P}, \boldsymbol{Q})$. We also use $\widehat{a}_{u,i,t}$ in place of $\widehat{a}^{\mathbf{tihad}}_{u,i,t}$ for brevity.

4.1 Optimization Formulation

Let $\boldsymbol{A}_1^T$ denote the set of all adoption labels in a diffusion cascade during the time span $[1, T]$.

$$\boldsymbol{A}_1^T \coloneqq \{a_{u,i,t} : t \in [1, T],\ u \in U \text{ and } i \in C_{u,t}\} \tag{12}$$

Diffusion data is then represented by a tuple of item set, the social network and the adoption labels as $\mathcal{D} = \left(I, G, \boldsymbol{A}_1^T\right)$. Given $\mathcal{D}$, we formulate the model learning problem as finding the optimal parameters $\boldsymbol{\Pi}^*$ that minimize squared error upon generating the adoption labels.
For a given $\boldsymbol{\Pi}$, the squared error at time step t is the sum

$$SE_t(\boldsymbol{\Pi}|\mathcal{D}) = \sum_{u \in U} \sum_{i \in C_{u,t}} \left[\widehat{a}_{u,i,t}(\boldsymbol{\Pi}) - a_{u,i,t}\right]^2 \tag{13}$$

Hence, over the whole time span $[1, T]$, the total error is

$$\mathcal{E}(\boldsymbol{\Pi}|\mathcal{D}) = \sum_{t=1}^{T} SE_t(\boldsymbol{\Pi}|\mathcal{D}) = \sum_{t=1}^{T} \sum_{u \in U} \sum_{i \in C_{u,t}} \left[\widehat{a}_{u,i,t}(\boldsymbol{\Pi}) - a_{u,i,t}\right]^2 \tag{14}$$

To avoid over-fitting, we also define a regularizer as

$$\mathcal{R}(\boldsymbol{\Pi}) \coloneqq h^2 + \sum_u b_u^2 + \sum_i b_i^2 + \|P\|_F^2 + \|Q\|_F^2 \tag{15}$$

where $\|.\|_F$ denotes the usual Frobenius norm. Hence, the objective function is

$$J(\boldsymbol{\Pi}|\mathcal{D}) = \frac{1}{2}\left[(\mathcal{E}(\boldsymbol{\Pi}|\mathcal{D}) + \delta\mathcal{R}(\boldsymbol{\Pi})\right]$$

We now can formulate the learning as the following constrained optimization problem.

Problem. *Given diffusion data set $\mathcal{D} = \left(I, G, \boldsymbol{A}_1^T\right)$. We learn parameters $\boldsymbol{\Pi}$ by solving for optimal parameters which minimize the objective function*

$$\boldsymbol{\Pi}^* = \underset{\boldsymbol{\Pi}}{\operatorname{argmin}}\, J(\boldsymbol{\Pi}|\mathcal{D}) = \underset{\boldsymbol{\Pi}}{\operatorname{argmin}}\, \frac{1}{2}\left[\mathcal{E}(\boldsymbol{\Pi}|\mathcal{D}) + \delta\mathcal{R}(\boldsymbol{\Pi})\right] \tag{16}$$

subject to constraints

$$\boldsymbol{p}_u \geq 0,\ \forall u \in U, \quad \boldsymbol{q}_i \geq 0,\ \forall i \in I \quad \textit{and}\ 0 \leq h \leq 1 \tag{17}$$

4.2 Optimization Solution

In general, the above problem is not convex. Thus, we resort to a solver which uses grid search and Projected Gradient Descent (PGD). For that, we provide formulae of gradients in the following sections. Due to space constraints, proofs of these formulae are not provided, interested readers can find it in the technical note.[1]

Derivatives for bias variables

$$\frac{\partial}{\partial \mu} J = \sum_{t} \sum_{u \in U} \sum_{i \in C_{u,t}} \overbrace{(\widehat{a}_{u,i,t}(\boldsymbol{\Pi}) - a_{u,i,t})}^{e_{u,i,t}} \tag{18a}$$

$$\forall u \in U,\ \frac{\partial}{\partial b_u} J = \delta b_u + \sum_{t} \sum_{i \in C_{u,t}} e_{u,i,t} \tag{18b}$$

$$\forall i \in I,\ \frac{\partial}{\partial b_i} J = \delta b_i + \sum_{t} \sum_{u \in U : i \in C_{u,t}} e_{u,i,t} \tag{18c}$$

Derivative for homophily variable

$$\frac{\partial}{\partial h} J = \delta h + \sum_{t} \sum_{u} \boldsymbol{p}_u^T \left[\boldsymbol{S}_t(u)\right] \boldsymbol{q}_t^{err}(u) \tag{19}$$

where $\boldsymbol{S}_t(u)$ is defined in Eqn. (10) and $\boldsymbol{q}_t^{err}(u) \coloneqq \sum_{i \in C_{u,t}} e_{u,i,t} \cdot \boldsymbol{q}_i$.

Derivatives for user and item factors

1. (Gradient w.r.t user factor $\boldsymbol{p}_u$) For each given user u, we have

$$\nabla_{\boldsymbol{p}_u} J = \delta \boldsymbol{p}_u + \sum_{t} \left[\boldsymbol{M}_t(u) \boldsymbol{q}_t^{err}(u) + h \eta_t(u) \boldsymbol{q}_t^{k}(u)\right] \tag{20}$$

   where matrix $\boldsymbol{M}_t(u)$ and scalar $\eta_t(u)$ are defined as

$$\boldsymbol{M}_t(u) \coloneqq \boldsymbol{Id} + h \boldsymbol{S}_t(u) \text{ and } \eta_t(u) \coloneqq \sum_{v \in N_u} \boldsymbol{p}_v^T \boldsymbol{q}_t^{err}(v) \tag{21}$$

   where $\boldsymbol{Id}$ denotes the identity matrix.
2. (Gradient w.r.t. item factor $\boldsymbol{q}_i$) For each given item i, we have

$$\nabla_{\boldsymbol{q}_i} J = \delta \boldsymbol{q}_i + \sum_{t} \left(h \sum_{u \in U} \left[\boldsymbol{q}_t^{err}(u) \boldsymbol{\varphi}_{u,i,t}^T\right] \boldsymbol{p}_u + \sum_{u : C_{u,t} \ni i} e_{u,i,t} \left[\boldsymbol{M}_t(u)\right]^T \boldsymbol{p}_u \right) \tag{22}$$

   where vector $\boldsymbol{\varphi}_{u,i,t} \coloneqq \sum_{recent\ adopters} \boldsymbol{p}_v$ is the sum of factors of neighbors who adopted i recently.

[1] http://goo.gl/2ltY9I

Algorithm 1 PGD for TIHAD model using an initial guess $\boldsymbol{\Pi}_0$

procedure TRAIN($\mathcal{D}$, $\boldsymbol{\Pi}_0$, ε)
  Initialize $\boldsymbol{\Pi}_c \leftarrow \boldsymbol{\Pi}_0$
  while $(!converge)$ **do**
    Compute objective value: $j_c \leftarrow J(\boldsymbol{\Pi}_c|\mathcal{D})$ ▷ use Eqns. (14) – (16)
    Compute gradients: $\boldsymbol{g}_c \leftarrow \nabla J(\boldsymbol{\Pi}_c|\mathcal{D})$ ▷ use Eqns. (18a) – (22)
    Descend & project: $\boldsymbol{\Pi}_n \leftarrow$ GRADPROJ($\boldsymbol{\Pi}_c, j_c, \boldsymbol{g}_c$) ▷ see `gradproj()` in [7]
    Check convergence: $converge \leftarrow (|\boldsymbol{\Pi}_n - \boldsymbol{\Pi}_c| < \varepsilon)$
    $\boldsymbol{\Pi}_c \leftarrow \boldsymbol{\Pi}_n$
  end while
  return $\boldsymbol{\Pi}_n$
end procedure

Now that all derivatives are available, we can use them in Projected Gradient Descent (PGD) with grid search to update the corresponding parameters. Thus, we repeat Algorithm 1 with different initial parameter values to learn the parameters of TIHAD model. All the derivatives in the algorithm are computed using Eqns. (18a) – (18c) and (19) – (22).

5 Experiments

In this study, we want to be able to evaluate TIHAD model with some parameter settings that control the item interaction and homophily factor during the diffusion process. Hence, we need a synthetic diffusion data generation method with the following input parameters: (a) M items, (b) N users, (c) N_e relationships among the users, (d) f latent factors, (e) homophily value h for the social network, (f) T number of time steps, and (g) k recently adopted items. The generation steps are described below:

1. (Generation of M items and N users in latent space) We generate M items and N users as f-dimensional vectors $\boldsymbol{q}_i$'s and $\boldsymbol{p}_u$'s respectively. The item and user vectors are generated such that each of them has a dominant factor. The set of users and items are denoted by U and I respectively.
2. (Generation of a social network with homophily value h) We generate N_e edges among the users using Algorithm 2. The resultant network, $G_h = (U, E_h)$ where E_h denotes the set of N_e edges, satisfies the required homophily level h.
3. (Generation of an initial adoption state) We want to ensure that every user in the network initially has adopted at least k items. We assign k items to each user based on his latent factor interests.
4. (Generation of a diffusion cascade) We randomly assign a user as the single seed of diffusion. The seed user will adopt all M items initially. We then employ TIHAD model to start generating a data set of simultaneous diffusion of the items over the network G_h within the time interval $[1, T]$. The details of this step are given in Algorithm 3.

We generate N diffusion cascades by performing steps 3 and 4 with a different initial adoption state and different user as the seed each time. Hence every diffusion

Algorithm 2 Generation of a network with a given homophily level

procedure BUILDNETWORK($U,\ N_e,\ h$)
  $Pairs \leftarrow \{(u, v) : u \neq v \in U\}$
  for each user pair $(u, v) \in Pairs$ **do**
    Compute user-item similarity: $sim(u, v) \leftarrow \boldsymbol{p}_u^T \boldsymbol{p}_v$
    Compute edge weight: $\rho(u, v) \sim \exp(h \cdot sim(u, v))$
  end for
  Normalize: $p(u, v) \leftarrow \frac{\rho(u,v)}{\sum \rho(u',v')},\ \forall (u, v) \in Pairs$
  Collect probabilities: $\boldsymbol{probs} \leftarrow (p(u, v) : (u, v) \in Pairs)$
  Sample N_e edges based on the probabilities: $E_h \leftarrow sample(Pairs, N_e, \boldsymbol{probs})$
return Network $G_h = (U, E_h)$
end procedure

Algorithm 3 Generation of diffusion data

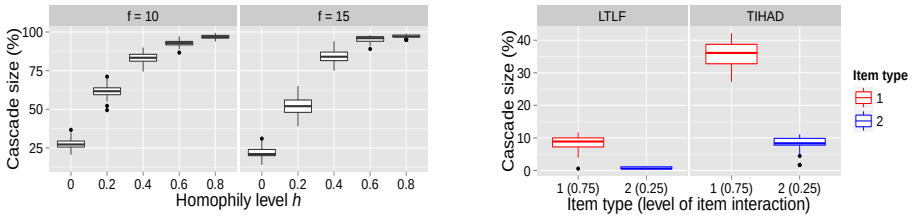
procedure CREATEDIFFUSION($I,\ G_h,\ \theta,\ T,\ u_s$) ▷ $G_h = (U, E_h)$: network in Algo. 2
  for $t \in [1, T]$ **do**
    Initialize $A_t \leftarrow \emptyset$ ▷ Set of adoption records at time t
    for $u \in U$ **do**
      Derive $C_{u,t}$ by Definition 1 ▷ use seed u_s to get $C_{u,1},\ \forall u$
      for $i \in C_{u,t}$ **do**
        Compute adoption label approximation $\widehat{a}_{u,i,t}$ by Eqn. (9)
      end for
      Pick adoptions $I_t(u) \leftarrow \{i \in C_{u,t} : \widehat{a}_{u,i,t} \geq 1 - \theta\}$ ▷ approx. is close to 1
      $A_t \leftarrow A_t \cup \{(u, i, t) : i \in I_t(u)\}$ ▷ Add to adoption records at time t
    end for
  end for
  Collect all adoption records: $\boldsymbol{A}_1^T \leftarrow \bigcup_{t=1}^{T} A_t$
  return $\mathcal{D} = \left(I, G_h, \boldsymbol{A}_1^T\right)$
end procedure

cascade share the same network with identical user and item latent factor vectors. We finally generate N different data sets so that we can get empirical distribution of cascade sizes.

5.1 Impact of Homophily on Diffusion

Experiment Setup: We study how the size of diffusion cascade is affected by different degrees of homophily h. Thus, we generate items and users by setting $f = 10$. We then generate diffusion in five different networks G_h's each with a different h value, $h \in \{0, 0.2, 0.4, 0.6, 0.8\}$. These networks however share the same set of users and same number of edges to minimize the effect of choices of users and number of relationships among them. For each such network, we generate N diffusion cascades of M items using TIHAD and study distribution of the average cascade size over the M items. Detailed statistics of this experiment is provided in Table 1.

Result: As the homophily level increases, the diffusion cascade also becomes larger (see Figure 1a). This trend is observed for all items. To evaluate the robustness of the



(a) Impact of homophily (cascades generated by TIHAD under different settings for number of factors f)

(b) Impact of item interaction (cascades generated by both models, for TIHAD we set parameters $h = 0.1$ and $f = 10$)

Fig. 1: Impact of homophily and item interaction on diffusion.

result, we repeat the experiment for $f = 15$ and $f = 20$. We report here results for $f = 10$ and $f = 15$. This result is expected as homophily facilitates diffusion ([5], [3]). It also shows that our model has incorporated homophily effect properly.

5.2 Impact of Item Interaction on Diffusion

Experiment Setup: In this experiment, we change our focus to study how item interaction (i.e. support among items) affects diffusion. We now generate diffusion cascades on the same network with a fixed homophily level $h = 0.1$. The item set is however generated differently. We partition the item set I into the *majority* set I_1 (occupy 75% of I) and the *minority* set I_2. In each subset, items are generated such that they are similar to each other. Thus, items in I_1 receive more interaction than items in I_2 and we can study difference in cascade sizes of items in two sets. Other statistics of this experiment is the same as in Table 1.

Under this setting, we use TIHAD model to simulate diffusion as done in the previous experiments. We then compare cascade size distribution of items in I_1 against that of items in I_2. We also want to see if cascades generated by TIHAD are significantly different from those generated by a baseline diffusion model that does not consider item interaction. Hence, we generate another set of cascades following the same process using the LTLF model. The cascade size distributions of the two models are then compared.

Result: Figure 1b shows several interesting insights. First, it provides strong evidence that TIHAD model can capture the item interaction effect (among similar items) currently ignored by the existing models including LTLF. The figure shows that the cascade size of an item diffused with TIHAD is much larger than that of the item when it is diffused using LTLF. Moreover, the more similar an item with previous items, the larger cascade size it can reach. This makes sense since an item will receive more support in diffusion if it is more similar to other previously adopted items.

Table 1: Parameters used in synthetic data generation

# factors	# items	# users	# edges	Homophily level	# recent items	# time steps
$f \in \{10, 15, 20\}$	100	500	$70K$	$h \in \{0, 0.2, 0.4, 0.6, 0.8\}$	$k = 5$	$T = 20$

Table 2: Statistics of diffusion data among Singapore Twitter users in Valentine Day

Data set	# hashtags	# users	# follow links	# adoptions	# time steps	# adoption labels
Training	4002	1000	9935	$11,565$	12	$60,875$
Test	1219	884	8754	9390	12	$39,375$
Total	4002	1000	9935	$20,955$	24	$100,250$

5.3 Hashtag Diffusion Prediction Evaluation

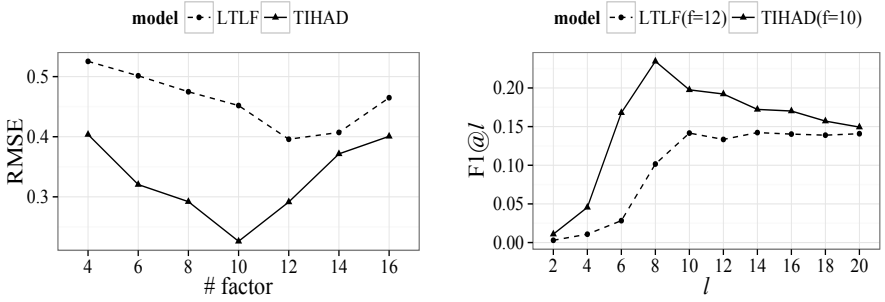
This experiment aims to evaluate TIHAD using real dataset and compare it with the baseline LTLF model which does not consider item interaction.

Data set: We first collected the diffusion of hashtags in the Twitter network among Singapore users during on 14 February 2014, the Valentine Day. We expected that there should be some interesting diffusion cascades on this special day. We extracted the tweets of about 150,000 Singapore users from 3 to 16 February and sampled 1000 active users who adopted at least 3 hashtags per day. These users are connected by a social network with 9935 follow links.

We next wanted to determine the time step when each user first adopted a hashtag during the Valentine Day. Each time step duration is set as one hour. We confined ourselves to *fresh* hashtags which only appeared during Valentine Day but not the days during [3 Feb, 13 Feb]. We then identified the time step a user adopted a hashtag as the first time step in 14 February he used the hashtag. We obtained $20,847$ hashtags which the active users adopted from 00:00am to 11:59pm on the Valentine day. By filtering away unpopular hashtags, i.e., those with less than 5 active users adopting them, we were left with 4002 hashtags and $20,955$ adoptions. Based on Definition 2, we derived $100,250$ adoption labels (both adoption and non-adoption) associated with these 24 hours. Adoptions of the users on previous day (13 Feb) were used as their initial adoption histories. The hashtag diffusion data on 14 February from 0:00am to 11:59am is then used as the training data, while the remaining data on 14 February is used as the test data. The statistics of combined training and test datasets is summarized in Table 2.

Training process: We trained both TIHAD and LTLF using the diffusion training dataset on February 14. We tried different values for the regularization constant and observed that $\delta = 0.1$ gives the best result in terms of minimizing RMSE. We also tried different values for the number of recent items $k \in \{1, \ldots, 10\}$ and found that $k \in \{3, 4\}$ yield the best RMSE result for this training dataset. In the learning process, we observed that both models can achieve smallest RMSE for the training data.

Evaluation metrics: For evaluations, we used two accuracy metrics: (i) RMSE for measuring the model performance during training, and (ii) $F1@l$ when using the trained models for the *hashtag adoption prediction task* on the test data. To compute $F1@l$, we use the trained models to predict hashtag adoptions (based on estimated adoption labels)



(a) RMSE obtained on training data

(b) $F1@l$ obtained on test data

Fig. 2: Comparing TIHAD against baseline LTLF. Both models were trained with regularization coefficient $\delta = 0.1$; for TIHAD, the number of recent items k is set as 3.

from 12:00 noon to 11:59pm of 14 Feb 2014. We selected those users who appear in both the training and test datasets and extracted from their tweets generated during the test period the hashtags that already appeared in the training set. The resultant test set had 884 users and 1219 hashtags which were actually adopted during the test period (detailed statistics of the test set can be found in Table 2).

Results: We first focus on the accuracy of trained models using RMSE defined on the training data. As shown in Figure 2a, the RMSE obtained by TIHAD is much smaller than that of LTLF when they are trained using the same dataset for different latent factor settings (i.e., $4 \leq f \leq 16$). TIHAD achieves the best RMSE when $f = 10$, while LTLF achieves best RMSE at $f = 12$.

In the prediction task, TIHAD shows a huge improvement over LTLF as shown in Figure 2b. Other than $l = 2$, TIHAD outperforms LTLF for all other i values. The highest $F1$ achieved by TIHAD ($F1@8$) is more than 150% that of LTLF ($F1@10$).

As TIHAD performs best for 10 factors, we would like to know what are the 10 factors. We manually check the top hashtags of each latent factor. We discover that the latent factors are topical and manually assign them topical labels. Table 3 shows the latent factors and their top 3 hashtags (due to limited space). Most of the latent factors (e.g., Music tour, Valentine, Electronics, Self-Improve) are self explanatory based on hashtags. The "Music bands/Singers" latent factor covers names of singers (e.g., Siti Nurhaliza and Eminem) and music concert (e.g., SUL14). The "Local movies/actors" latent factor covers popular movies (e.g., "You Who Came From the Stars", "Brothers Keeper") and actor (e.g., Gong Li). The other latent factors can be interpreted in a similar manner.

Finally we would like to see what TIHAD can tell us about the network based on the homophily level and influence weights it learned. The homophily level learned by TIHAD is $h = 0.08$. This value is quite small and can be explained due to the sparseness of the network under study. Moreover, the histogram of influence weights

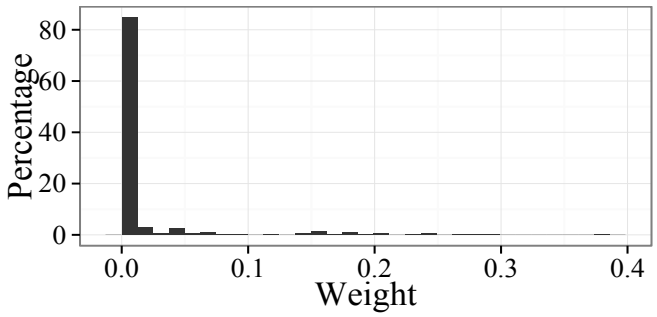


Fig. 3: Histogram of influence weights $w_{v,u}$ which TIHAD learned for the network of Twitter users in our experiment.

Table 3: Latent factors and their top-3 hashtags

Latent Factors	Hashtags
Music bands/Singers	eminemftw, DatoSitiNurhaliza, SUL14
Local movies/actors	YouWhoCameFromTheStars, BrothersKeeper, GongLi
International movies/actors	frozen, jimmyfallon, KristenWiig
Music tour	RedAsiaTour, TheScriptUSTour, BANGERZTour2014
Sport	ICC2014, F1NightRace, LFCfacebook
Beauty	ILoveWTF, Dior, maybellinesg
Valentine	happyvalentine, firstvalentine, TweetforLove
Scandal/Controversy	AsylumSeekers, bigimmigrationrow, LittleIndiaRiot
Electronics	Xiaomi, ipadmini, Logitech
Self-improve	limitless, nickvijucic, empoweryourself

$w_{v,u}$ in Figure 3 shows that most weights are very small (80% of them are close to 0), which matches the nature of weak links among most Twitter users.

6 Conclusion

This work deals with the challenging problem of modeling multiple simultaneous diffusion processes where topic level interaction exists among items being diffused in a social network with homophily. We successfully incorporate item interaction and homophily by proposing a novel way to model social influence from recent adoptions of user's neighbors. Behavior of the model under different settings and parameters have been investigated. Results on synthetic data show that both homophily and interaction at topic level can increase diffusion remarkably. Experiment on hashtag diffusion on Twitter shows that TIHAD can model interacting diffusion effectively and give better prediction as well.

Since training TIHAD is not a convex problem, we are currently using grid search to deal with the non-convexity. However, the problem is still convex for each set of

parameters if others are kept fixed. Thus, we plan to use Alternating Descent to develop a more rigorous algorithm.

Acknowledgement. This research is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office, Media Development Authority (MDA).

References

1. Wei Chen, Yifei Yuan, and Li Zhang. Scalable influence maximization in social networks under the linear threshold model. In *Proceedings of the 10th ICDM*, pages 88–97. IEEE, 2010.
2. Nicholas A Christakis and James H Fowler. The spread of obesity in a large social network over 32 years. *New England journal of medicine*, 357(4):370–379, 2007.
3. Munmun D. Choudhury, Hari Sundaram, Ajita John, D. Duncan Seligmann, and Aisling Kelliher. "birds of a feather": Does user homophily impact information diffusion in social media? *arXiv:1006.1702*, 2010.
4. Julien Delporte, Alexandros Karatzoglou, Tomasz Matuszczyk, and Stéphane Canu. Socially enabled preference learning from implicit feedback data. In *Proceedings of ECMLPKDD*, pages 145–160, 2013.
5. Benjamin Golub and Matthew O Jackson. How homophily affects the speed of learning and best-response dynamics. *The Quarterly Journal of Economics*, 127(3):1287–1338, 2012.
6. Mark Granovetter. Threshold models of collective behavior. *American journal of sociology*, pages 1420–1443, 1978.
7. Carl T Kelley. *Iterative methods for optimization*, volume 18. Siam, 1999.
8. David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the 9th KDD*, pages 137–146. ACM, 2003.
9. Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th KDD*, pages 426–434, 2008.
10. Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
11. Shuyang Lin, Qingbo Hu, Fengjiao Wang, and Philip S Yu. Steering information diffusion dynamically against user attention limitation. In *Proceedings of the 14th ICDM*, 2014.
12. Hao Ma, Haixuan Yang, Michael R Lyu, and Irwin King. Sorec: social recommendation using probabilistic matrix factorization. In *Proceedings of the 17th CIKM*, pages 931–940. ACM, 2008.
13. Andriy Mnih and Ruslan Salakhutdinov. Probabilistic matrix factorization. In *Proceedings of the 20th NIPS*, pages 1257–1264, 2007.
14. EM Rogers. *Diffusion of innovations.* New York (USA), Free Press, 1983.
15. Ruslan Salakhutdinov and Andriy Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th ICML*, pages 880–887. ACM, 2008.
16. Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th WWW*, 2001.
17. Yelong Shen and Ruoming Jin. Learning personal+social latent factor model for social recommendation. In *Proceedings of the 18th KDD*, pages 1303–1311. ACM, 2012.
18. Xiaoyuan Su and Taghi M Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009:4, 2009.
19. L. Weng, A. Flammini, A. Vespignani, and F. Menczer. Competition among memes in a world with limited attention. *Scientific Reports*, 2012.