Learning the Influence Probabilities Based on Multipolar Factors in Social Network

Weibo Wang, Zhaohui Peng(✉), Ziyan Liu, Tianchen Zhu, and Xiaoguang Hong

School of Computer Science and Technology, Shandong University, Jinan, China
wangweibo1001@gmail.com, {pzh,hxg}@sdu.edu.cn,
liuziyan1990@163.com, ztc1319@sina.cn

Abstract. How to model the influence propagation accurately in social network is a critical and challenge task. Although numerous attempts have been made for this topic, few of them consider the user's negative influence. Positive influence will encourage people to perform some action while the negative one will degrade the probability. Thus, it is meaningful to model the influence propagation by considering both the positive and negative influence. What's more, previous research is mostly based on the assumption that the influence probabilities between users are known, however, they are typically unknown in real-world social networks. To address these problems, a novel Multipolar Factors aware Independent Cascade model (MFIC) is proposed to outline the information diffusion in social network. Then, the user-to-user influence probability is learnt with the users' behavior logs based on the EM algorithm. We also apply the discovered influence probabilities to user behavior prediction. Experiments are conducted over real data sets, Flixster and Digg, validating the effectiveness of our methods.

Keywords: Social network · Multipolar influence · Influence probabilities

1 Introduction

The social networks such as Twitter, Digg and Flixster, providing platforms for people to share information and express their ideas, play an important role in information diffusion. Much attention has been paid to the research on social influence and influence-driven information diffusion in social network, which have been applied in many areas, such as viral marketing [1], product recommendation [2, 3] and user behavior prediction [4]. For example, in viral marketing application, if a seller wants to promote a new product under a limited budget, he will choose some users with high influence in the social network and give them free products to use, and then, by the cascade effects produced by word-of-mouth, more people will be driven to buy this product.

So far, a substantial research effort has been dedicated to develop more accurate propagation model [5,6] based on the Independent Cascade mode(IC) and Linear Thread model(LT) . However, these work only consider the positive influence between users. In addition to the positive influence, there is also negative influence

S. Zhang et al. (Eds.): KSEM 2015, LNAI 9403, pp. 512–524, 2015.
DOI: 10.1007/978-3-319-25159-2_46

between users. For example, in Flixster[1], the users will rate the movies they have seen, as shown in Fig.1. Given a movie *i*, the u_1 ' neighbors u_2, u_3, u_4 gave high scores (4.*5, 4.8, 4.7*), while the neighbors u_5, u_6, u_7 gave low scores (*2.5, 2.3, 1.5*). When u_1 makes a decision to see the movie or not, he may see the rating scores of his friends. It can be plausibly concluded that u_2, u_3, u_4 have a positive influence to u_1, because their high ratings tend to promote u_1 to see the movie and u_5, u_6, u_7 have a negative influence to u_1, because their low ratings tend to decrease the probability of u_1 seeing the movie. So it is important to model the influence propagation by considering both the positive and the negative factors.

Moreover, some conventional studies in social influence [1, 7, 8] arbitrarily assume the social network has edges labeled with the probability that a user's action will be influenced by his neighbor's behaviors. However, the influence probabilities are typically unknown in real-world social network. Despite previous work [9, 10] have studied how to estimate the influence probabilities in social network, a key limitation is their ignoring of negative influence aforementioned above.

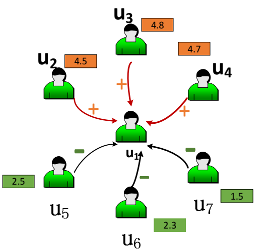

Fig. 1. An example of positive influence and negative influence

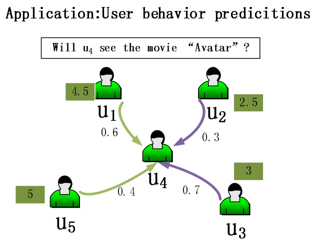

Fig. 2. An application of our work.

In this research, our goal is to address the issues above: we propose a novel model that modeling the information diffusion through analyzing the multipolar influence. And then, based on our proposed model, we learn the influence probabilities. We also introduce the application of user behavior prediction based on the learnt influence probability. For example, as depicted in Fig.2, in the time-step *t*, u_4's neighbors u_1, u_5 gave high rating scores on the movie "Avatar", while the neighbors u_2, u_3 gave low rating scores, and then we can predict whether u_4 will see the movie in the time-step *t+1* based on the influence probabilities computed by our method. To summarize, this work contributes on the following aspects:

1. A novel Multipolar Factors aware Independent Cascade model (MFIC) is proposed to model the influence propagation in social network. In MFIC, we analyze the behaviors of users by considering both the neighbors' positive and negative influence on him.

---

[1] http://www.flixster.com/

2. We design a method based on the EM algorithm to learn the parameters in our model. In our method, we use the social relationship and the users' past behavior logs to learn the influence probabilities between users based our MFIC model. We also apply the discovered influence probabilities to user behavior prediction.
3. Experiments are conducted on two real data sets: Flixster and Digg. Experimental results show that the learnt influence probabilities based on our MFIC model can greatly improve the accuracy of user behavior prediction.

The rest of the paper are organized as follows: Section 2 formally formulates the foundations for our problem. Section 3 explains the proposed model MFIC and Section 4 introduces the method of parameter learning in detail. In Section 5, we introduce the application of user behavior prediction. In Section 6, we experimentally compare and evaluate our model with other models. Finally, Section 7 discusses the related work and Section 8 concludes the work.

2 Preliminaries

2.1 Independent Cascade (IC) Model

Independent Cascade (IC) model [1] is one of the widely used representative influence diffusion model. In the IC model, given a network *G (V, E)*, for each directed link $e= (u, v) \in E$, we specify a value $p_{u,v}(0<p_{u,v} < 1)$. Here $p_{u,v}$ is the influence propagation probability from *u* to *v*. The diffusion process starts with some initial active nodes (called "seeds") and proceeds in the following way: when a node *u* first becomes active at time-step *t*, it has only one chance to activate its each current inactive out-neighbor *v* and the attempt succeeds with the probability $p_{u,v}$. If the attempt succeeds, *v* becomes active at time *t+1*. The attempt is performed only at time-step *t*, whether or not *u* succeeds and *u* will not make any further attempts to activate *v* in the subsequent rounds. The process terminates until no more nodes can be activated.

2.2 Problem Formulation

Definition 1 (Social Network). A social network can be represented as *G= (V, E)*, where *V* denotes the set of users, *E* is the set of edges. A directed / undirected edge $(u,v) \in E$ represents a social link between user *u* and user *v*. In some social networks like Twitter and Digg the edge is directed which represents *v* has followed *u* and *u* will influence *v* while in Flixster and Facebook the edge is undirected which represents they are friends for each other and they will influence each other.

Definition 2 (User Behavior Log). The user behavior log Ω is a set of actions (*User, Item, Time*), which a tuple $(u, i, t_u) \in \Omega$ indicates that user *u* performs an action for item *i* at time t_u. We assume that no user performs the same action more than once. The projection of Ω on *User* is contained in the set of nodes *V* of the social network.

Definition 3 (Positive Neighbor and Negative Neighbor). For an edge $(u,v) \in E$, if u is active on item i at time t, we denote as $u_i^t = 1$, that is to say, u performs the action on item i at time t. If u has a positive opinion on the product, such as giving a high rating score to the movie, we think u is a positive user as well as a positive neighbor of v, $u \in N_v^{positive}$ and u will have a positive influence to v. Otherwise, if u has a negative opinion on the product, such as giving a low rating score to the movie, we think u is a negative user as well as a negative neighbor of v, $u \in N_v^{negative}$ and u will have a negative influnce to v.

3 MFIC: Multipolar Factors Aware Independent Cascade Model

In the proposed MFIC model, the working principle is similar to Independent Cascade model [1]. The diffusion process unfolds in discrete time-steps t and begins from a given initial active user set. When a user v observes a piece of information at time t, he makes his decision depending on his neighbor's status. If he adopts the information, his status becomes active at time $t+1$, otherwise inactive. For example, we can imagine the information is a movie in Flixster and user adopts a movie means he saw the movie. In time t, some of his neighbors saw the movie, which we think whether the user (u_1, as shown in Fig.1) will see the movie is influenced by both his positive neighbors(u_2,u_3,u_4) and negative neighbors(u_5,u_6,u_7). If u_1 sees the movie, we regard u_1 as influenced successfully by the positive neighbors or failure influenced by his negative neighbors and becoming active. Otherwise, if u_1 doesn't see the movie, we regard u_1 as influenced successfully by the negative neighbors or failure influenced by his positive neighbors and become inactive.

In each time-step $t+1$, the user v receives two kinds of influence, one is the positive influence effected by the positive neighbors $N_{positive,t}$(v) in time-step t, and another is the negative influence effected by the negative neighbors $N_{negative,t}$(v) in time-step t. We also assume that the probabilities of different neighbors influencing the user are independent, each neighbor has a probability to trigger the user to perform the action or not. In time-step $t+1$, the influence that user v receives from the positive neighbors denotes as $p_v^{positive}(t+1)$ and the influence that user v receives from the negative ones denotes as $p_v^{negative}(t+1)$. Their calculation formula are follows:

$$P_v^{positive}(t+1) = 1 - \prod_{u \in N_{positive,t}(v)} \left(1 - p_{u,v}\right) \tag{1}$$

$$P_v^{negative}(t+1) = 1 - \prod_{u \in N_{negative,t}(v)} \left(1 - p_{u,v}\right) \tag{2}$$

For example, in Flixster, if user u sees the movie i, we denote $s_i(u)=1$ and $r_i(u)$ represents the rating score that user u give to the movie i. We use D_i(t) represent the users that saw the movie at time t. In reality, if the user very like the movie, he will give a high rating score to it, and if he dislike, he will give a low rating score. We use the avg(i) denotes the average score of the movie i. If u sees the movie i and the rating score $r_i(u)$ >avg(i), we think u is a positive user at time t, $u \in D_{positive}$(t), and u is a positive neighbor of v, i.e., $u \in N_{positive,t}$(v). Otherwise, u gives a low rating score, i.e., r_i(u) <avg(i), we think u is a negative user at time t, u$\in D_{negative}$(t) , and u is a negative neighbor of v, $u \in N_{negative,t}$(v).

In time-step $t+1$, user v will become active if the positive neighbors successfully influence v or the negative neighbors failure in influence v, and the probability that user v becomes active can be computed as follows.

$$\begin{aligned}\bar{p}_v^{active}(t+1) &= P_v^{positive}(t+1) + \left(1-P_v^{negative}(t+1)\right) - P_v^{positive}(t+1) * \left(1-P_v^{negative}(t+1)\right) \\ &= 1 - p_v^{negative}(t+1) + p_v^{positive}(t+1) * p_v^{negative}(t+1)\end{aligned} \quad (3)$$

Equally, when the negative neighbors successfully influence v or the positive neighbors failure in influence v, user v will become inactive, and the probability of being inactive as following Eq.(4).

$$\begin{aligned}\bar{p}_v^{inactive}(t+1) &= P_v^{negative}(t+1) + \left(1-P_v^{positive}(t+1)\right) - P_v^{negative}(t+1) * \left(1-P_v^{positive}(t+1)\right) \\ &= 1 - p_v^{positive}(t+1) + p_v^{positive}(t+1) * p_v^{negative}(t+1)\end{aligned} \quad (4)$$

To unified comparison, we normalize the $\bar{p}_v^{active}(t+1)$, $\bar{p}_v^{inactive}(t+1)$ as follows.

$$P_v^{active}(t+1) = \frac{\bar{p}_v^{active}(t+1)}{\bar{p}_v^{active}(t+1)+\bar{p}_v^{inactive}(t+1)}$$

The equation of $\bar{p}_v^{inactive}(t+1)$ is similar to $\bar{p}_v^{active}(t+1)$.

Following Saito *el al.* [9], we also assume the input propagation have the same shape as they were generated by the MFIC model itself. This means that the propagation trace of an item i must be a sequence of sets of users D_i(0),…, D_i(n), corresponding to the discrete time steps of the MFIC propagation. Moreover for each node $v \in D_i$(t+1), there exists at least a neighbor u of v such that $u \in D_i(t)$. Next, let B_i(t) denote a set of users having become active by time-step t, $B_i(t) = \bigcup_{t'<t} D_i(t')$. Let use $C(u)$ denotes the child nodes of u:$C(u)=\{v|(u,v) \in E\}$, $F(v)$ denotes the parent nodes of v: $F(v)=\{u|(u,v) \in E\}$. We use D_i denote the propagation trace of item i, for an propagation trace D_i we can define the following likelihood function as a joint probability of every observed user status on item i on every timestamps with respect to $\theta = \{p_{u,v}\}$ in Equation (5).

$$L(\theta, D_i) = \left(\prod_{t=0}^{T-1} \prod_{v \in D(t+1)} \left\{ P_v^{active}(t+1) \right\}\right)\left(\prod_{t=0}^{T-1} \prod_{u \in D(t)} \prod_{v \in C(u) \setminus B(t+1)} \left\{ P_v^{inactive}(t+1) \right\}\right) \quad (5)$$

There are many items propagation in the network, so we let $\{D_i : i = 1, \dots, I\}$ be a set of independent information diffusion episodes. Then we can define the following object function with respect to θ.

$$L(\theta)=\sum_{i=1}^{I}\log L(\theta,D_i)$$
$$=\sum_{i=1}^{I}\sum_{t=0}^{T-1}\left\{\left(\sum_{v\in D_i(t+1)}\log P_v^{\text{active},i}(t+1)\right)+\left(\sum_{u\in D_i(t)}\sum_{v\in C(u)\setminus B_i(t+1)}\log P_v^{\text{inactive},i}(t+1)\right)\right\}$$
$$=\sum_{i=1}^{I}\sum_{t=0}^{T-1}\left[\left\{\sum_{v\in D_i(t+1)}\left(\log[(1-P_v^{negative,i}(t+1))+P_v^{\text{positive},i}(t+1)*P_v^{negative,i}(t+1)]\right)\right\}\right] \tag{6}$$
$$+$$
$$\sum_{i=1}^{I}\sum_{t=0}^{T-1}\left[\left\{\sum_{u\in D_i(t)}\sum_{v\in C(u)\setminus B_i(t+1)}\left(\log[(1-P_v^{positive,i}(t+1))+P_v^{negative,i}(t+1)*P_v^{positive,i}(t+1)]\right)\right\}\right]$$

where $P_v^{active,i}(t+1)$, $P_v^{inactive,i}(t+1)$ represent the probability of v become active or inactive about item i at time-step $t+1$. $P_v^{posotive,i}(t+1)$, $P_v^{negative,i}(t+1)$ stand for the probability that user v was affected by the positive influence or negative influence successfully at time-step $t+1$ about movie i, and their computational formulas as given in Eq.(1)-Eq.(4). Then our problem is to obtain the set of influence probabilities between users, $\theta=\{p_{u,v}\}$, which maximizes Eq.(6). In the next section we will illustrate how to obtain the parameters.

4 Learning the Parameters of MFIC

Directly maximizing Equation (6) is rather not tractable, so we apply the Expectation Maximization(EM) algorithm [13] to obtain the parameters $\theta=\{p_{u,v}\}$. In the rest of the paper, following the standard EM notation, $\hat{p}_{u.v}$ will represent the current estimate of the influence probability of user u to user v.

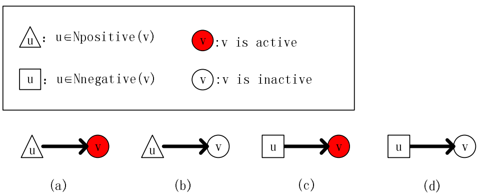

Fig. 3. The cases among link *(u, v)* for item *i.*

For a link *(u, v)* in the propagation trace D_i of item i where $u \in D_i(t)$, we know that the user u will attempt to influence v with the probability $\hat{p}_{u.v}$. There are four cases existing among link *(u, v),* as shown in Fig. 3. For case (a), user u is a positive neighbor of user v, and user v became active at time step $t+1$, which means user v was successfully influenced by the positive influence or failure influenced by the negative influence, so the probability that user v was activated by user u is $\hat{p}_{u.v}/\hat{p}_v^{active,i}$ and v was activated not because of u with the probability (1- $\hat{p}_{u.v}/\hat{p}_v^{active,i}$), where $\hat{p}_v^{active,i}$ is calculated by using Eq.(3). On the other hand, as is shown in case (b), user v is not active at time step $t+1$, so we can be surely think the attempt that user u try to activate user v failed. Similar, for case (c), user u is a negative neighbor of user v and user v is active at time step $t+1$, and we can be surely think the attempt that user u try to make user v inactive failed. For case(d), user v is inactive at time step $t+1$, so the probability that user v was inactivated by user u is $\hat{p}_{u.v}/\hat{p}_v^{inactive,i}$ and v was inactivated not because of user v with the probability (1- $\hat{p}_{u.v}/\hat{p}_v^{inactive,i}$), where $\hat{p}_v^{inactive,i}$ is calculated by using Eq.(4). Considering these case, we have the following *Q-function* describe users' status for all propagation traces $D_i = \{i|\ i = 1,\dots,I\}$

$$
\begin{aligned}
Q(\theta|\hat{\theta}) = & \sum_{i=1}^{I}\sum_{t=0}^{T-1}\sum_{u\in D_{i,positive}(t)}\left(\sum_{v\in C(u)\wedge v\in D_i(t+1)}\left\{\frac{\hat{p}_{u.v}}{\hat{p}_v^{active,i}}\log p_{u,v}+\left(1-\frac{\hat{p}_{u.v}}{\hat{p}_v^{active,i}}\right)\log(1-p_{u,v})\right\}+\sum_{v\in C(u)\backslash B_i(t+1)}\left\{\log(1-p_{u,v})\right\}\right) \\
+ & \sum_{i=1}^{I}\sum_{t=0}^{T-1}\sum_{u\in D_{i,negative}(t)}\left(\sum_{v\in C(u)\backslash B_i(t+1)}\left\{\frac{\hat{p}_{u.v}}{\hat{p}_v^{inactive,i}}\log p_{u,v}+\left(1-\frac{\hat{p}_{u.v}}{\hat{p}_v^{inactive,i}}\right)\log(1-p_{u,v})\right\}+\sum_{v\in C(u)\wedge D_i(t+1)}\left\{\log(1-p_{u,v})\right\}\right)
\end{aligned}
\tag{7}
$$

Let $\partial Q/\partial p_{u,v} = 0$, obtaining the new estimate of $p_{u,v}$, the update equation is following:

$$
p_{u,v} = \frac{1}{|P_{u,v}^{+}|+|P_{u,v}^{-}|+|N_{u,v}^{+}|+|N_{u,v}^{-}|}\left(\sum_{i\in P_{u,v}^{+}}\frac{\hat{p}_{u,v}}{\hat{p}_v^{active,i}}+\sum_{i\in N_{u,v}^{+}}\frac{\hat{p}_{u,v}}{\hat{p}_v^{inactive,i}}\right) \tag{8}
$$

Here $P_{u,v}^{+}$ denotes the items that u successfully influence v by positive influence, which satisfies both $u\in D_{i,positive}(t)$ and $v \in D_i(t+1)$. $P_{u,v}^{-}$ denotes the items that u failure influence v by positive influence, which satisfies both $u\in D_{i,positive}(t)$ and $v \notin D_i(t+1)$. $N_{u,v}^{+}$ denotes the items that u successfully influence v by negative influence, which satisfies both $u\in D_{i,negative}(t)$ and $v\notin D_i(t+1)$. $N_{u,v}^{-}$ denotes the items that u failure influence v by negative influence, which satisfies both $u\in D_{i,negative}(t)$ and $v \in D_i(t+1)$. Moreover, $|P_{u,v}^{+}|$, $|P_{u,v}^{-}|$, $|N_{u,v}^{+}|$, $|N_{u,v}^{-}|$ denote the number of items in them .

Our Expectation-Maximization method for learning the parameters of MFIC is given in Algorithm 1. The learning algorithm takes input the social graph $G=(V,E)$ and a log of past propagations. The output is the set of all parameters θ: those are $p_{u,v}$ for all the edge $(u, v) \in E$. The learning method starts with a random initialization of the probabilities of all the edges with value $p_{u,v} \in (0\ 1)$ (line1). We know that the EM algorithm is related with the initial value, that is different initial parameters will bring different locally optimal solution, so we set various values in our experiments, and we find that the initial values set between 0.6 and 0.9 could get the best effects. Then for each edge $(u,v) \in E$ finding the items that u influenced v successfully by positive influence or negative influence, and we compute the probability of user v becomes active or inactive in these items (line3-line18), which equals the E-step in EM. And then, for each edge $(u,v) \in E,$ update the probability $p_{u,v}$ of user u influence user v using the Equations(8) (line19-line 21), which equals the M-step in EM. Finally, the process will end until the change of the probabilities between two times converge to a threshold. The EM is an iterative updating algorithm, which will update every parameter in every iteration. So when there are a lot of arguments, the running time will become longer.

Algorithm 1. EM method of learning the parameters of MFIC

Input: Social graph $G=(V, E)$, User behavior log Ω.
Output: The set of all parameters of MFIC θ, that is: $\forall (u, v) \in E$: $p_{u,v}$

1. init $\{p_{u,v}\}$
2. repeat
3. **For** all the $(u,v) \in E$ **do**
4.   **For** every item i in Ω
5.   **If**($u \in N_{positive,t}$(v))
6.   **If** (v is active)
7. $P_{u,v}^{+} = P_{u,v}^{+} \cup \{i\}$
8.   Compute $P_v^{active,i}(t+1)$
9.   **Else**
10. $P_{u,v}^{-} = P_{u,v}^{-} \cup \{i\}$
11.   **If**($u \in N_{negative,t}$(v))
12.   **If** (v is active)
13. $N_{u,v}^{-} = N_{u,v}^{-} \cup \{i\}$
14.   **Else**
15. $N_{u,v}^{+} = N_{u,v}^{+} \cup \{i\}$
16.   Compute $P_v^{inactive,i}(t+1)$
17.   **End For**
18.   **End For**
19.   **For** every the $(u,v) \in E$ **do**
20. $p_{u,v} = \frac{1}{|P_{u,v}^{+}|+|P_{u,v}^{-}|+|N_{u,v}^{+}|+|N_{u,v}^{-}|}\left(\sum_{i \in P_{u,v}^{+}} \frac{\hat{p}_{u.v}}{\hat{p}_v^{active,i}} + \sum_{i \in N_{u,v}^{+}} \frac{\hat{p}_{u.v}}{\hat{p}_v^{inactive,i}}\right)$
21.   **End For**
22.   until convergence;

5 User Behavior Prediction

The learned influence probabilities among users can be used to help with many applications. Here we illustrate one application on user behavior prediction, i.e., how the learned influence can improve the performance of user behavior prediction.

Based on the MFIC model proposed in Section 3, we present the Algorithm 2 for predicting the user behavior. This algorithm focuses on the question of whether a user will perform a behavior at time-step $t+1$, given the behaviors of his neighbors at time-step t. For example, in Flixster, the behavior is defined as whether a user rates a movie and in Digg, the behavior is defined as whether a user digs a story. For a user u, if he performed the behavior, we think that user u is active. Otherwise, we think that u is inactive. For each item i and inactive user v in the testing dataset, we find his positive and negative neighbors in time-step t and calculate the user's the positive and negative influence receives from his positive and negative neighbors (line 3- line 6). Then we compute the probability that user be active and inactive (line 8-line 9). If the probability that user be active is larger than user be inactive, we think the user become active, otherwise, inactive (line 10- line 12).

Algorithm 2. User Behavior Prediction

Input: Social network $G=(V,E)$, User behavior log Ω, influence probabilities $\{P_{u,v}\}$
Output: The user's state for the item i in the testing dataset

1. **For** each item i in the testing dataset
2. **For** each inactive user v
3. Find positive neighbors of v: $N_{positive}(\mathrm{v})$
4. $P_v^{positive} = 1 - \prod_{u \in N_{positive}(\mathrm{v})} (1 - p_{u,v})$
5. Find negative neighbors of v: $N_{negative}(\mathrm{v})$
6. $P_v^{negative} = 1 - \prod_{u \in N_{negative}(\mathrm{v})} (1 - p_{u,v})$
7. **End For**
8. $P_v^{active} = (1 - P_v^{negative}) + P_v^{positive} * P_v^{negative}$
9. $P_v^{inactive} = (1 - P_v^{positive}) + P_v^{positive} * P_v^{negative}$
10. **If** $P_v^{active} \geq P_v^{inactive}$
11. Then v is active;
12. Else v is inactive;
13. **End For**

6 Experiments

In this section, we report our results on two real datasets and we compare our MFIC model to the state-of-the-art models. Our goal is to validate whether our proposed model can help to describe real-world influence cascade.

6.1 Datasets

The datasets in our experiments are Flixster[2] and Digg[3]. They are publicly available, both containing a social graph G=(V,E) and a set of past propagation log $\Omega = \{(\text{User}, \text{Item}, \text{Time})\}$. Next we describe the data sets in the following:

Table 1. Details of the Flixster, Digg datasets

Statistics	Flixster		Digg	
	Training	Test	Training	Test
#Users	15,675	5,104	27,488	18,664
#Items	8,105	4613	3553	2786
#Actions	1,433,768	480,000	2,517,067	414,620
#Friendship	1,084,895	250,096	683,160	492,138

[2] http://www.cs.sfu.ca/~sja25/personal/datasets/
[3] http://www.isi.edu/~lerman/downloads/digg2009.html

Flixster. Flixster is one of the main players in the mobile and social movie rating business. In this context, the action is defined as user rate the movie, if user *u* gives a high score on the movie, we think a positive influence will happen between user *u* and user *v*. otherwise, we think a negative influence will happen between *u* and *v*.

Digg. Digg is a social network website, where users vote stories. In this context, if user *u* votes a story, we think *u* have a positive influence on *v*. If *u* didn't vote the story, but at least one of his neighbors did, we think *u* will impose negative influence on *v*.

We preformed some standard consistency cleaning on the two datasets. We remove those items that appear in the log Ω less than 20 times. We also remove those users that not appear in the log Ω and have no friends. Moreover, for the experiment we perform a chronological split of log Ω in both datasets into training (80%) and testing (20%). Details of datasets are shown in Table 1.

6.2 Experimental Setup

For different datasets, the life span of information is various, as shown in Fig. 4, in Digg (the left part), most behaviors occurred in the first 40 hours, so we set the time-step interval at 5, 10, 15, 20, 25, 30 hours respectively that divide the users in the dataset into different time-step. In Flixster (the right part), most behaviors occurred within the first 36 months, so we set the time-step interval at 4, 8, 12, 16, 20, 24 months.

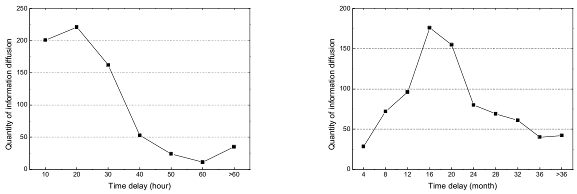

Fig. 4. The information diffusion quantity over time delay in Digg and Flixster.

We apply the learned influence probabilities for user behavior prediction as described in Section 5. We compare the following methods to our proposed PNIC model and evaluate its performance in terms of Precision, Recall and F1-Measure.

- PIC. The PNIC model which only consider the positive factor without the negative factor, the influence probabilities among users also are learnt.
- Static Model. Static model is the method proposed in [10], since we don't focus on the time-dependent influence propagation in this paper, so we only compare our method with the Static Model. The influence probability $p_{v,u}$ is computed by Equation (9), where $|I_{v,u}|$ is the number of actions that *v* has influenced *u* and $|I_v|$ is the number of actions performed by *v*.

$$p_{v,u} = \frac{|I_{v,u}|}{|I_v|} \tag{9}$$

- IC. The influence probability for each edge (u, v) is assigned as 0.01, which is widely adopted by previous studies with the IC model.

6.3 Prediction Performance Analysis

Fig. 5 and Fig. 6 show the prediction performances of all the tested approaches under different measurements at different time-step interval on Digg and Flixster dataset. We can see that the proposed PNIC model can consistently achieve better performance comparing with baseline methods, the IC model worst. Notably, both PNIC and PIC all perform better than Static Model (with an improvement 2-6%). Because in Static Model, the influence probability $p_{u,v}$ is computed only based the number of information diffusions from u to v. And in IC model, the influence probabilities among users are random assigned. Therefore, the predicting performances of Static Model and IC model are uncompetitive. In contrast, in PNIC and PIC model, the influence probabilities are learnt by the user behavior log and considering all the interacting users, so improve the performance significantly. The experiment results confirm that our model considering both positive and negative influence will better describe real-world.

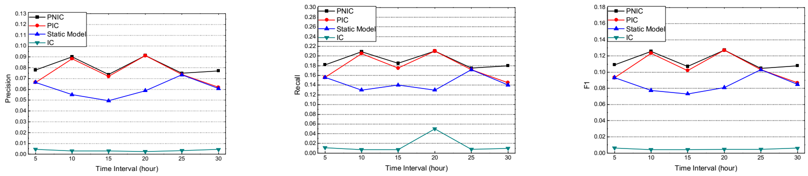

Fig. 5. Prediction performances on Digg dataset.

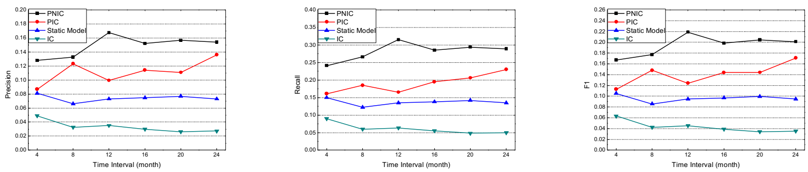

Fig. 6. Prediction performances on Flixster dataset.

Furthermore, we also can see the performance of our PNIC model outperforms the PIC in Flixster, significantly improvements (3-10%). While in Digg they perform similar, but still has an improvement 1.6% in some case. Our explanation is that in Flixster, when a user wants to see a movie, the opinion of his friends are very important. If the most friends gave a low rate, he may don't see the movie. So the negative influence plays an important role in user behavior. But in Digg, vote or not is a very easy action, so when he see his friends vote the story he may possible to vote it even though other friends didn't vote. We can conclusion that the importance of negative influence in Flixster is larger than in Digg. It is very useful when analyze the user behavior in different social network.

7 Related Work

The problem of influence propagation in social network has been widely studied [1, 7, 8]. While pervious works on influence maximization typically assume a social graph G with edges labeled with influence probabilities. Very few works focus on estimating the influence probabilities without the characteristic of items (such as the contents of Twitter). The most relevant works with us are [9, 10]. Satio *et al.* [9] focus on the IC model and defined the likelihood for multiple episodes. They present a method for predicting diffusion probabilities from a log of past propagations by using the EM algorithm. Bonchi *et al.* [10] devise various probabilistic models of influence and develop algorithms for learning the influence probabilities. However, none of them pay attention to the influence probabilities calculation by considering the negative influence.

To the best of our knowledge only few papers have analyzed social influence considering the negative factor [11, 12]. Li *et al.* [11] quantifies the influence and conformity of each individual in a network by utilizing the positive and negative relationships between individuals. However, they don't propose any propagation model, nor study the user-to-user influence probability. Chen *et al.*[12] discuss the influence diffusion considering the negative influence, but their focus is design efficient heuristic for influence maximization in social network rather than learning the influence probability.

8 Conclusion and Future Work

In this paper, a novel influence propagation model MFIC is proposed to model the information diffusion incorporating the positive and negative influence. We model the user's behavior consider the multipolar factors. Then, we design method to learn the influence probabilities based on the history behavior logs of users and we also apply the discovered influence probabilities to user behavior prediction. We conduct experiments to test the effectiveness of our model in real datasets. In future work, we will take advantage of the influence probabilities to design more accurate model to predict user's behaviors considering other factors, such as the user preference.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (Grant No.61303005, No.61170052), the Natural Science Foundation of Shandong Province of China (Grant No.ZR2013FQ009), the Science and Technology Development Plan Project of Shandong Provience (Grant No.2014GGX101047, No.2014GGX101019).

References

1. Kempe, D., Kleinberg, J., Tardos, E.: Maximizing the spread of influence through a social network. In: KDD, pp. 57–66, August 2003
2. Huang, J., Cheng, X.Q., Shen, H.W., et al.: Exploring social influence via posterior effect of word-of-mouth recommendation. In: WSDM, pp. 573–582, February 2012
3. Ye, M., Liu, X., Lee, W.C.: Exploring social influence for recommendation: a generative model approach. In: SIGIR, pp. 671–680, August 2012
4. Tang, J., Sun, J., Wang, C., Yang, Z.: Social influence analysis in large-scale networks. In: KDD, pp. 807–816, June 2009
5. Barbieri, N., Bonchi, F., Manco, G.: Topic-aware social influence propagation models. In: ICDM, pp. 81–90 (2012)
6. Li, D., Xu, Z., Luo, Y., Li, S.: Modeling information diffusion over social networks for temporal dynamic prediction. In: CIKM, pp. 1477–1480, October 2013
7. Chen, W., Wang, C., Wang, Y.: Scalable influence maximization for prevalent viral marketing in large-scale social networks. In: KDD, pp. 1029–1038, July 2010
8. Chen, W., Wang, Y., Yang, S.: Efficient influence maximization in social networks. In: KDD, pp. 199–208, June 2009
9. Saito, K., Nakano, R., Kimura, M.: Prediction of information diffusion probabilities for independent cascade model. In: Lovrek, I., Howlett, R.J., Jain, L.C. (eds.) KES 2008, Part III. LNCS (LNAI), vol. 5179, pp. 67–75. Springer, Heidelberg (2008)
10. Goyal, A., Bonchi, F., Lakshmanan, L.: Learning influence probabilities in social networks. In: WSDM, pp. 241–250, February 2010
11. Li, H., Bhowmick, S.S., Casino, S.A.: Towards conformity-aware social influence analysis in online social networks. In: CIKM, pp. 1007–1012, August 2011
12. Chen, W., Collins, A., Cummings, R., et al.: Influence maximization in social networks when negative opinions may emerge and propagate. In: SDM, pp. 379–390, April 2011
13. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society **39**, 1–38 (1977)