Rewarding Social Recommendation in OSNs: Empirical Evidences, Modeling and Optimization

Li Ye, *Member, IEEE*, Hong Xie, *Member, IEEE*, Yishi Lin, and John C. S. Lui, *Fellow, IEEE*

Abstract—In the past few years, many companies are considering "*social recommendation*" for their businesses, e.g., firms are offering rewards to customers who recommend the firms' products/services in online social networks (OSNs). However, the pros and cons of such social recommendation scheme are still unclear. Thus, it is difficult for firms to design rewarding schemes, and for OSN platforms to design regulating policies. By analyzing real data from Weixin and Yelp, we first identify key factors that affect the spreading of products/services in OSNs. These findings enable us to develop an accurate (i.e., with a high validation accuracy) mathematical model on social recommendations. Our model captures how users decide whether to recommend an item, which is a key factor but often ignored by previous social recommendation models such as the "*Independent Cascade model*". We also design algorithms to infer model parameters. Using our model, we uncover conditions when social recommendation improves a firm's profit and users' utilities, as well as when it cannot improve the profit or hurts users' utilities. These conditions help the design of both rewarding schemes and regulating policies. Moreover, we extend our model to a dynamic setting, so that a firm can improve its profit by dynamically optimizing its rewarding schemes.

Index Terms—Incentive scheme, referal marketing, user-to-user recommendation, percolation theory, data analytics, profit maximization

1 Introduction

With the prevalence of online social networks (OSNs) such as Facebook and Twitter, products can reach a large number of customers via friend-to-friend recommendations (called social recommendations). To encourage social recommendations, a number of firms are offering non-monetary or even monetary reward. For example, Dropbox gives extra storage space to users who recommend friends[1] to use Dropbox [12]. Some businesses in Yelp, e.g., restaurants, give gifts to customers who do *"check-in"* in the social network [24].

In this work, we consider an incentivized social recommendation scheme in which a user receives rewards from firms by recommending firms' products/services to friends. To illustrate the tradeoff for firms and the platform, consider the following three examples:

Example 1 (Baseline case). Consider the baseline case where the firm gaves no reward to recommenders, i.e., scenario 1 in Fig. 1. Three users form a social network as a line-graph. The production cost of a product is \$3 and the firm sets a price of \$7. The firm used a traditional advertisement (Adv.) in which only user 1 was informed. She purchased the product. But she would not recommend it to her friends in the OSN because there was no reward. User 2 and 3 would not be informed of the product. The firm's total profit was (\$7−\$3)+\$0+\$0=\$4.

Example 2 (Benefits of social recommendation). Consider social recommendation with rewards, i.e., scenario 2 in Fig. 1. The firm offered a reward of \$4 to users who recommend the product to their friends. The firm increased the price from \$7 to \$9 to compensate the cost of reward. With such reward, user 1 would recommend the product to user 2, so user 2 knew the product from *"friends' recommendations"* (Rec.). User 2 purchased it but decided not to recommend it, and user 3 was not informed of the product. The firm's total profit was (\$9−\$3−\$4)+(\$9−\$3)+\$0=\$8, which was higher than the profit of \$4 in scenario 1 without rewards to recommenders.

Example 2 highlights that the firm can *improve* the profit by rewarding social recommendations. However, such scheme has certain risk and the following example illustrates it.

Example 3 (Potential risks of social recommendation). Consider the scenario 3 in Fig. 1, where the firm now increases the reward to \$5. Then, both user 1 and user 2 are willing to recommend the product. User 3 thinks the price is too high and decides not to buy it. Therefore, the firms's total profit is \$2, which is smaller than the profit, i.e., \$4, of not rewarding social recommendation. Worse yet, user 3 could be annoyed as she receives an uninteresting recommendation.

Example 3 highlights that social recommendation may potentially lead to profit loss to the firm. Futhermore, some users may receive uninteresting recommendations, which is a threat to the OSN's platform and its eco-system.

1. Here, a user's "friend" can be anyone who can receive information from the user, e.g., subcribers in online social network or email receiver.

- *Li Ye and John C. S. Lui are with the Chinese University of Hong Kong, Hong Kong. E-mail: liye666666@gmail.com, cslui@cse.cuhk.edu.hk.*
- *Hong Xie is with the Chongqing University47913, Chongqing 400044, China. E-mail: xiehong2018@foxmail.com.*
- *Yishi Lin is with Tencent508929, Shenzhen 518054, China E-mail: yishilin14@gmail.com.*

Manuscript received 3 October 2019; revised 10 August 2020; accepted 22 October 2020. Date of publication 18 November 2020; date of current version 5 August 2022.
(Corresponding author: Hong Xie.)
Recommended for acceptance by G. Koutrika.
Digital Object Identifier no. 10.1109/TKDE.2020.3038930

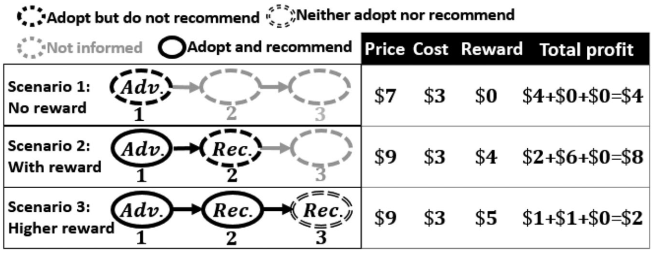


Fig. 1. Profitability and tradeoffs of social recommendation.

The above examples motivate us to answer the following important questions:

- *For social recommendation in OSNs, what are the key factors that affect firms' profit gains and users' utilities?*
- *How much reward/price to set so to increase profit?*

Answering the above questions is challenging. First, real-world social network consists of millions of users and the topology is more complicated than the line graph in Fig. 1. Second, when the price and reward changes, firms do not know consumers' behaviors and the potential profit gains. In Fig. 1, it is possible that in scenario 2, user 2 thinks the price is too high and does not buy it. In this case, the firm will have a total profit of $2, which is less than the baseline profit of $4. One can see that users' uncertain behaviors can greatly affect the firm's profits. Third, the OSN platform does not know users' utilities on the recommended items. We address these questions and our contributions are:

- *Novel model with data:* With large datasets of Weixin and Yelp, we conduct an in-depth empirical analysis of the spreading pattern of social recommendations, and we uncover number of key factors, e.g., users' recommendation rate. These findings enable us to develop an accurate mathematical model on social recommendations. In particular, our model captures how users decide whether to recommend an item, which is often ignored by previous social recommendation models such as the "Independent Cascade model"[14]. Compared with the Independent Cascade model, we reduce the error to predict the number of recommenders by more than 80 percent on Weixin's dataset.
- *Novel findings:* By both theoretical analysis and trace-driven simulations, we reveal the pros and cons of social recommendations: (1) a firm can improve profits using social recommendations when its item is not well-known, or users' recommending rate is near a *"critical value"*, etc.; (2) although social recommendation improves the utilities of the users who recommend the items, it often hurts non-recommenders' utilities especially for OSNs whose users have weak ties with friends (e.g., Twitter), etc.
- *Novel algorithm to improve firms' profits:* We design an efficient algorithm to estimate a firm's profit, and it is more than 6,000 times faster than agent-based simulations [10], making it easier to search for a firm's optimal strategy. We also extend our model to an dynamic setting and design algorithms for a firm to dynamically optimize rewarding schemes. Compared with influence maximization algorithms [23] that are not explicitly aware of firms' profits, our algorithm improves a firm's profit by as high as 71 percent on a Facebook's graph.

TABLE 1
Statistics of Social Recommendation Campaigns (in Weixin)

Item ID	marketing activity	reward	discount	recommend rate	click rate
1	*share*	**1**	0	**4.56%**	**2.1%**
2	*share*	1	0	4.56%	1.3%
3	*click*	0.05	0	2.64%	1.2%
4	*new-user*	1	0	0.95%	4.0%
5	*new-user*	0.5	0	0.82%	2.3%
6	*discount*	0	0.1	0.19%	0.9%
7	*discount*	0	1	2.91%	0.8%
8	*no-reward*	0	0	0.76%	6.4%
9	*no-reward*	0	0	0.54%	4.5%

The reward and discount are normalized to [0,1]. E.g., the reward "0.05" means the firm gives a 5 percent discount.

In Section 2, we analyze the data of rewarded social recommendations in Weixin and Yelp. In Section 3, we propose a model to capture the important factors found by data analysis. In Section 4, we theoretically characterize the spreading patterns of social recommendations and the firm's optimal rewarding strategy. In Section 5 we show how to infer the model parameters, and we use the inferred parameters to simulate the social recommendations in Section 6. In Section 7, we extend the settings and design algorithms to allow a firm to dynamically optimize its strategy.

2 Evidences From Real-World Social Recommendation Data

In this section, we analyze real-world datasets from Weixin and Yelp. Through this we find that the users' recommending rate is critical to the spreading of a firm's product in OSNs. We also observe various factors, e.g., reward, information source, product price, etc., which significantly influence users' decisions on whether to recommend an item. These observations help us to build an accurate mathematical model in Section 3.

2.1 Evidences From Weixin Data

The Weixin Data. Weixin is an OSN in China with more than one billion monthly active users [2]. Our dataset was collected from volunteers and is fully anonymized. There are three types of rewards for the recommenders: reward the customers (1) who *"share"* an item to his receivers for a certain number of times; or (2) whose receivers *"click"* the item shared by him; or (3) who successfully invite a *"new user"* to adopt an item. In addition, a user can get *"discounts"* when he buys an item via the received referral link. Also, an item can have no reward and no discount, which case is abbreviated as *"no-reward"*. Therefore, an item has one of the five marketing activities: *share, click, new-user, discount* and *no-reward*. The volunteers decide whether to share or click the item based on the rewards. We collected statistics in the social recommendation campaigns among these volunteers to quantify rewards and discounts, and normalize to [0,1].

Table 1 presents an example of data for a firm with multiple products (items). For item 1, a recommender can get a reward of 1. Among the users who obtain *"at least one"* recommendations of item 1, 4.56 percent of them further

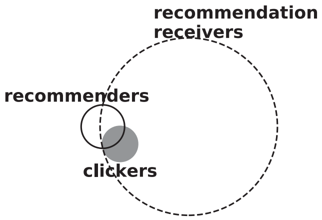


Fig. 2. Recommenders, recommendation receivers and clickers for all items.

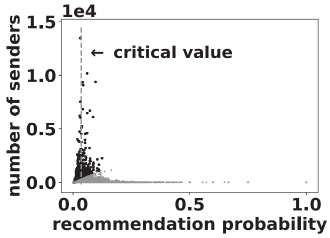


Fig. 3. Recommendation probability and the fraction of senders.

recommend the item to their receivers. Moreover, 2.1 percent of the users who receive a recommendation of item 1 would click the item. Note that a user can receive a recommendation of item 1 for multiple times. This is why we see the recommendation rate 4.56 percent is greater than the click rate 2.1 percent. For each item, our dataset contains the set of recommenders (i.e., anonymized users who recommended the item) and the set of receivers (i.e., anonymized users who received recommendations of the item). These anonymous volunteers form an undirected social network. Due to privacy restrictions, we only use aggregated statistics such as the degree distribution to create the social network.

• *Data analysis 1: uncover the spreading pattern of an item.* We study the spreading pattern of an item by investigating the number of receivers that obtain recommendations about the item. Fig. 4a shows the impact of the fraction of recommenders on the fraction of receivers, where each point corresponds to one item. One can observe that the fraction of receivers increases *nearly linearly* in the fraction of recommenders with a slope of 15.6. Fig. 4b further shows how the fraction of senders in the user population is imapcted by the users' recommending probability (or rate), which is defined as

$$\textit{recommendation probability} \triangleq \frac{\text{num. of receivers who recommend it}}{\text{num. of receivers of an item}}. \tag{1}$$

One can observe that there is a threshold (we call it the "*critical value*") on the recommendation probability, below which the fraction of senders is small and above which the fraction of senders is large.

To further validate the diffusion pattern of social recommendations, we also analyze the data for 8,799 articles of Weixin's official accounts, where Weixin's official accounts typically do not provide rewards to recommenders. Fig. 3 plots the relationship between the recommendation probability and the fraction of senders. Generally, the fraction of senders is large only when the recommendation probability is close to or greater than the critical value. But we see a large recommendation probability does not imply the a large number of senders. The reason might be that when the number of recommendation receivers is small, we only calculate the recommendation probability of a small group of users, which has a high variance and lead to a large value of recommendation probability. We use the *gray* color for the articles whose fraction of recommendation receivers is below 0.04 percent of the user population. One can see that when we filter out those *gray* points, a high recommendation probability implies a large fraction of senders in the population.

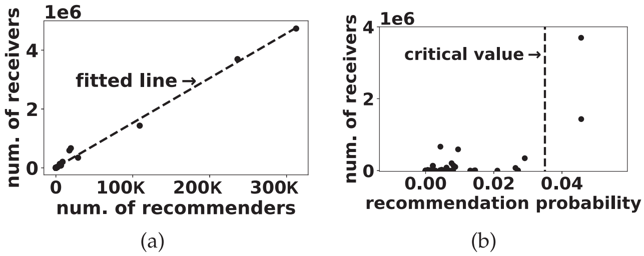


Fig. 4. Impact of *(a) the fraction of recommenders, (b) recommendation probability* on the fraction of senders.

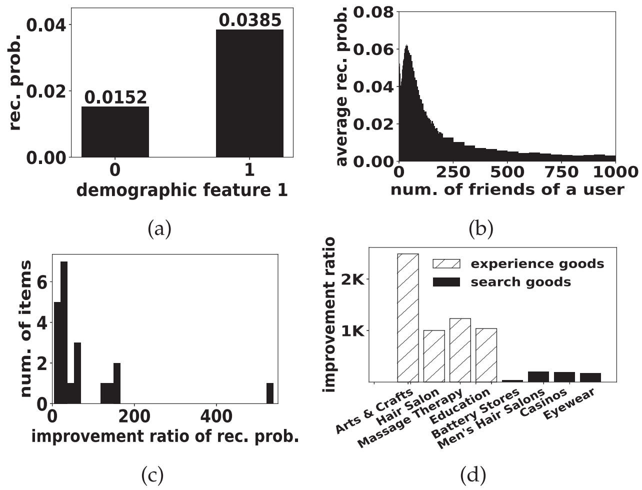


Fig. 5. Impact of *(a) user's demographic feature; (b) degree of nodes; (c) information source, (d) the perceivability of item's value* on the recommendation prob.

Moreover, we study the overlap of users who receive recommendations and users who send recommendations. Fig. 2 shows that a majority of recommenders are also receivers of recommendations. Since the receivers of recommendations also send a large number of recommendations, the cascade of recommendations cannot be ignored.

All the above analyses implies that the *recommendation probability* is critical for the information spreading of an item, and we will further study it next.

• *Data analysis 2: uncover key factors that influence recommendation probability.* Fig. 5a shows the recommendation probability varies significantly for a demographic feature. Fig. 5b shows that as the degree (num. of receivers) of a user increases, the recommendation probability first increases and then decreases. In particular, users with 35 receivers have the highest recommendation probability. To see how the information from recommendations improves recommendation probability, we note that a user can know an item from other sources (say, conventional media). We define

TABLE 2
Impact of Reward on Recommendation Probability

	With reward (average)	Without reward (max.)
Rec. prob.	**0.0271** ($> 3.5 \times 0.0076$)	0.0076

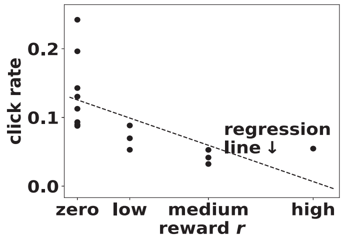


Fig. 6. [Weixin] Rewards and click rates (one point for one firm's items).

$$improvement\,ratio\,of\,recommendation\,probability \triangleq \frac{\mathbb{P}[\text{recommend the item}|\text{receive recommendation}]}{\mathbb{P}[\text{recommend the item}|\text{do not receive recommendation}]} - 1. \tag{2}$$

Fig. 5c is the histogram of items' improvement ratios. We see that the information of recommendations always improves users' recommendation probability. The improvement ratio can reach two orders of magnitude. Table 2 shows that on average, offering reward can increase users' recommendation probability by 250 percent compared with users' recommendation probability without reward. We point out that the reward affects not only a user's recommendation probability but also the user's adoption probability, as one would also consider the reward as a kind of discount in buying an item. We will model this in Section 3.

• *Data analysis 3: users' satisfiability for social recommendation.* We use click rate to measure users' satisfiability. In Fig. 6, we use the item with the highest number of recommenders as the representative of each firm. We observe that users' click rates are lower for items that are associated with rewards, compared to those with zero reward. According to the regression line (where the rewards for "zero", "low" "medium", "high" are set to 0, 0.2, 0.5, 0.9), we see that users' click rates and rewards are negatively correlated.

Moreover, to reduce the impact of an item's category on users' click rate, we further compare the click rates of three items in the same category in Table 3. We observe that the click rate of the item without rewards is significantly higher than that of items with rewards. We need to be cautious about this correlation. The problem is that the observations only imply correlations but not causal relationships. For example, the high click rate for an item might be caused by this item's inherent characteristics, but not the associated reward. To see whether the observed correlation is from some confounding factors, on the same data as that for Fig. 6 we conduct another regression analysis considering the price p, recommendation probability q for users who receive recommendations, and recommendation probability $\tilde{q}$ for users who do not receive recommendations. Denote the reward as r, then a more complicated linear regression model is

$$\text{click rate} = -0.127r + 0.0653p + 0.293q - 53.6\tilde{q} + 0.0471.$$

TABLE 3
The Click Rate of 3 Items in the Same Category

	1st item	2nd item	3rd item
provide reward?	Yes	No	Yes
click rate	0.052826	**0.1963**	0.06968

TABLE 4
(Yelp) Number of Check-in versus Number of Reviews

	# of businesses	(1) = # of check-ins	(2) = # of reviews	ratio (1)/(2)
w/ check-in offer	6,238	1,086,014	329,006	**3.30**
w/o check-in offer	79,236	8,225,098	2,737,303	**3.00**

Compared with the baseline linear regression model "click rate $= -0.132r + 0.125$" in Fig. 6, we notice that the regression coefficient of reward is similar. This implies that the correlation between the reward and the click rate is robust under different confounding factors.

Lessons Learned. The recommendation probability for an item needs to be above a critical value so that the item's information can widely spread in an OSN. A number of factors, e.g., users' characteristics, reward, information source, influence the recommendation probability significantly. We observe that users' click rates become lower when firms offer reward for recommendations. There are a number of important factors, e.g., price, product quality, etc., that may influence the recommendation probability but are not covered in the Weixin dataset. We next study them via a dataset from Yelp.

2.2 Evidences From Yelp's Data

The Yelp Data. The dataset is from the 11th round of Yelp's dataset challenge [25]. This dataset contains an unweighted undirected social network of 1,326,101 users and 85,474 businesses. We treat check-ins and reviews as recommendations since friends are notified. Businesses provide rewards for customers who do check-in, i.e., "check-in offer"[24]. Table 4 summarizes the statistics of reviews and check-ins of the dataset. We observe that users' check-ins are correlated to reviews, as the ratios between the total number of check-ins and the total number of reviews are similar for businesses that provide or do not provide check-in offer.

• *Data analysis 4: more factors that influence recommendation probability.* Based on whether the quality of an item is perceivable before purchasing, we classify items into experience goods (one cannot perceive the quality before purchasing, e.g., medical services) and search goods (one can perceive the quality before purchasing, like houses). Fig. 5d shows that the improvement ratios of recommendation probability for experience goods are much higher than that for search goods.

Lessons Learned. When an item's quality is hard to perceive before purchasing, friends' recommendations greatly increase a user's recommendation probability because she trusts her friends' recommendations.

3 Modeling Social Recommendation

In this section, we present a model that captures all the observations in Section 2. First, we present a reward

TABLE 5
Table of Notations

$\mathcal{N}$	the set of N users
$\mathcal{I}$	the set of I items
$\mathcal{G} = (\mathcal{N}, \mathbf{W})$	weighted directed graph for the OSN
$\mathbf{W}$	the weight matrix for the graph
m, n	the index for users
i	the index for item or the firm of the item
r_i	the reward to do recommendation task for firm i
p_i	price for a unit of item i
c_i	marginal cost to sell a unit of item i
S_{ni}	information status of user n on item i
$\emptyset, \mathcal{O}, \mathcal{R}, \tilde{\mathcal{O}}, \tilde{\mathcal{R}}$	notations for the information status (Fig. 7)
v_{ni}	item i's valuation for user n
u_{ni}	item i's intrinsic utility for user n
$\hat{u}_{ni}$	the estimated utility of item i for user n
γ_i	perceivability of item i's valuation
$Q_{ni}(S_{ni})$	personalized quality of item i for user n
τ	user n's trust score on friends' recommendations
$\tilde{\tau}$	user n's trust score on other information sources
g_{ni}	user n's utility gain to recommend item i
$\tilde{v}_{ni}$	social value for user n to recommend item i
$\tilde{c}_n$	cost to do the recommendation task for user n
A_{ni}/R_{ni}	whether user n adopt/recommend item i
$U_{ni}(A_{ni}, R_{ni} \mid S_{ni})$	estimated utility for decision (A_{ni}, R_{ni})
δ_i	a user's probability to know item i without friends' recommendation
$a_{ni}(S_{ni})$ / $q_{ni}(S_{ni})$	user n's probability to adopt/recommend item i
$D_i(p_i, r_i)$	expected # of users who adopt item i (demand)
$R_i(p_i, r_i)$	expected # of users who recommend item i
$P_i(p_i, r_i)$	"net profit" for the firm of item i

scheme for social recommendations. Second, we present users' decision model, capturing factors on recommendation probability. Third, we formulate a firm's problem to maximize its profit. Table 5 summarizes key notations of this paper.

3.1 The Rewarding Scheme and Firm's Decision Space

We consider a market, which consists of $N \in \mathbb{N}_+$ users denoted by $\mathcal{N} \triangleq \{1, \ldots, N\}$ and $I \in \mathbb{N}_+$ items (or products) denoted by $\mathcal{I} \triangleq \{1, \ldots, I\}$. An item could be the service of Dropbox, the newest iPhone, etc. Each item i is provided by one firm and we call it "firm i". In the case that a firm sells multiple items, we treat it as multiple "virtual" firms. The OSN is modeled as a weighted directed graph $\mathcal{G} \triangleq (\mathcal{N}, \mathbf{W})$ among users, where $\mathbf{W} \triangleq [w_{mn}: m, n \in \mathcal{N}] \in [0, 1]^{N \times N}$. A directed link from user m to n captures that user m can recommend items to user n. The weight w_{mn} quantifies the influence strength of user m on user n. A larger w_{mn} indicates a stronger influence.

The firm i can post a reward scheme [12] to incentivize users to recommend items to their friends, which consists of: (1) a recommendation task associated with item i; (2) the reward $r_i \in \mathbb{R}_+$ for completing the task. For example, the task for Dropbox is to invite a certain number of new users, and the reward is a maximum volume of 22 GB cloud storage space. Only the users who adopt (or use) the item are allowed to accept the task. Each user can adopt at most one unit of item and accept at most one task. The firm i sets a price of $p_i \in \mathbb{R}_+$ for item i, which has a per unit marginal cost of $c_i \in \mathbb{R}_+$. The decision for the firm i is to jointly select the price and reward (p_i, r_i) for each item i, where $p_i \geq 0, r_i \geq 0$.

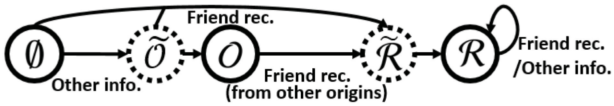


Fig. 7. Transitions of a user's information state.

3.2 The Users' Decision Model

We consider two types of information sources that influence users' decision, i.e., recommendation and other information sources (e.g., traditional advertisements). We track user n's information status on item i using a state variable $S_{ni} \in \{\emptyset, \mathcal{O}, \mathcal{R}, \tilde{\mathcal{O}}, \tilde{\mathcal{R}}\}$, where $S_{ni} = \emptyset, \mathcal{O}$ or $\mathcal{R}$ means that user n does not know item i, is informed of it from conventional information sources (e.g., TV or newspaper advertisement), or is informed of it from friends' recommendations respectively. To describe the diffusion of recommendations, we have two additional states $\tilde{\mathcal{O}}$ and $\tilde{\mathcal{R}}$ for a user. The state $S_{ni} = \tilde{\mathcal{O}}$ (or $\tilde{\mathcal{R}}$) means that user n will be informed of item i from other information sources (or friends' recommendations), but are not currently informed. For example, a user is in $\tilde{\mathcal{R}}$ when he receives a recommendation but does not click it. Fig. 7 depicts the state transition, and these states have a total order $\emptyset \prec \tilde{\mathcal{O}} \prec \mathcal{O} \prec \tilde{\mathcal{R}} \prec \mathcal{R}$, representing the strength of information can only evolves from weak to strong. A user's behaviors depend on her states. This captures the impacts of information sources on users' behaviors (Fig. 5c). In the above state transition, a user will be in state $\mathcal{R}$ if at least one friend recommends the item to him. We do not add more states to count the number of friends who recommend the item to a user (which is similar to assumptions in Independent Cascade model [14]).

When a user is in state $\mathcal{O}$ or $\mathcal{R}$, he makes decisions. Let $v_{ni} \in \mathbb{R}_+$ denote the valuation of item i by user n, which is the highest price that user n is willing to pay for item i. The intrinsic utility for user n to buy item i or u_{ni}, is

$$u_{ni} \triangleq v_{ni} - p_i, \qquad \forall n \in \mathcal{N}, i \in \mathcal{I}. \tag{3}$$

The u_{ni} represents the personalized quality of item i for user n.

Before adopting an item i, user n can only *"estimate"* u_{ni} partially based on the true value u_{ni} and partially based on her information source (represented by S_{ni}). We denote the estimated utility as $\hat{u}_{ni}(S_{ni})$ and model it as

$$\hat{u}_{ni}(S_{ni}) \triangleq \gamma_i u_{ni} + (1 - \gamma_i) Q_{ni}(S_{ni}), \tag{4}$$

where $\gamma_i \in [0, 1]$ models the perceivability of item i, and $Q_{ni}(S_i) \in \mathbb{R}$ denote the personalized quality summarized from a user's information sources. A larger γ_i models that an item is more perceivable in its value (or quality). For example, the haircut service has a small perceivability γ_i while a house has a large γ_i. Let $\tau_n, \tilde{\tau}_n \in \mathbb{R}$ denote user n's trust on recommendations and other information sources respectively. The trust reflects how a user thinks an item known from some information source is worthy of its price. We model Q_{ni} as

$$Q_{ni}(S_{ni}) \triangleq \begin{cases} \tau_n p_i, & \text{if } S_{ni} = \mathcal{R}, \\ \tilde{\tau}_n p_i, & \text{if } S_{ni} = \mathcal{O}. \end{cases} \tag{5}$$

We require $\tau_n \geq \tilde{\tau}_n$ to capture that users often have higher trust on their friends' recommendations. By modeling

	$S_{ni}{=}\mathcal{O}$	$S_{ni}{=}\mathcal{R}$
Class 1	(1,1)	(1,1)
Class 2	(0,0)	(1,1)
Class 3	(0,0)	(1,0)
Class 4	(1,0)	(1,0)
Class 5	(0,0)	(0,0)

Fig. 8. User's decision (A^*_{ni}, R^*_{ni}) in different states: 5 classes.

item's perceivability and users' trust, we capture previous observation in Fig. 5d, i.e., friends' recommendations have different effects on users' behaviors for items with different perceivability.

We have modeled a user's utility to adopt an item. Now we define a user n's utility gain to recommend an item i as

$$g_{ni} \triangleq r_i + \tilde{v}_{ni} - \tilde{c}_n, \qquad \forall n \in \mathcal{N}, i \in \mathcal{I}, \tag{6}$$

where $\tilde{v}_{ni}{\in}\mathbb{R}$ denotes the social value and $\tilde{c}_n{\in}\mathbb{R}_+$ denotes the cost to take up a recommendation task. Recommending good items enhances friendship, leading to a large social value, while recommending a poor item leads to a small social value.

A user n's decision is to choose whether to adopt and recommend item i denoted by $A_{ni}{\in}\{0,1\}$ and $R_{ni}{\in}\{0,1\}$ respectively, where $A_{ni}{=}1$ (0) means user n adopts (or not) and $R_{ni}{=}1$ (0) means user n recommends (or not). Recall that a user can take up a recommendation task if and only if she adopts an item. Thus, the decision space for user n with respect to item i is $(A_{ni}, R_{ni}){\in}\{(0,0),(1,0),(1,1)\}$. We define the estimated utility for decision (A_{ni}, R_{ni}) as

$$U_{ni}(A_{ni}, R_{ni}|S_{ni}) \triangleq \begin{cases} 0, & \text{if } (A_{ni}, R_{ni}){=}(0,0), \\ \hat{u}_{ni}(S_{ni}), & \text{if } (A_{ni}, R_{ni}){=}(1,0), \\ \hat{u}_{ni}(S_{ni}){+}g_{ni}, & \text{if } (A_{ni}, R_{ni}){=}(1,1). \end{cases}$$

A user decides whether to adopt or recommend an item, when she is informed of the item (her state changes to $\mathcal{O}$ or $\mathcal{R}$). We assume her decision only depends on her estimated utility $U_{ni}(A_{ni}, R_{ni}|S_{ni})$. This *"utility-driven"* assumption is not restrictive, since the utility captures factors such as the item's reward (Table 2), the item's category (Fig. 5d), user's features (Figs. 5a and 5b), the item's price, etc. For example, when user n maximizes her estimated utility, the optimal decision is

$$(A^*_{ni}, R^*_{ni}){\in} \arg \max_{(A_{ni},R_{ni}){\in}\{(0,0),(1,0),(1,1)\}} U_{ni}(A_{ni}, R_{ni}|S_{ni}).$$

Recall that once a user has adopted or recommended an item, the adoption or recommendation can not be revoked. Also, a user only adopts or recommends an item for at most one time. After adopting an item, whether a user will recommend an item is independent of the information status, so a user n's action R^*_{ni} should be same in state $\mathcal{O}$ and $\mathcal{R}$. Thus, users can be categorized into five classes based on their actions under different states, as shown in Fig. 8. For example, both class-2 users and a class-3 users neither adopt nor recommend in state $\mathcal{O}$, and adopt in state $\mathcal{R}$. But class-2 users recommend in state $\mathcal{R}$, while class-3 users do not.

3.3 The Firm's Decision Model

We divide time into slots starting from $t{=}1$. Let us focus on item i for descriptions. At the beginning of time slot 1, each user is in state $\tilde{\mathcal{O}}$ with probability δ_i. Namely, every user can know item i from other information with probability δ_i afterwards. When a user gets informed of an item (from any sources), we say the user *"arrives"* at the firm. At the beginning of time slot $t>0$, one of the users in states $\tilde{\mathcal{O}}$ or $\tilde{\mathcal{R}}$ arrives. Then, the arrived user makes decisions as described in Section 3.2. If user m decides to recommend, then with probability w_{mn}, a user n will be informed of the item and transit to state $\tilde{\mathcal{R}}$. Recall that w_{mn} is the influence strength of user m on user n. It captures that a user may receive recommendations from (and be influenced by) close friends.

To illustrate, consider the example in Fig. 9. At the beginning of time slot 1, user 3 arrives because he is in state $\tilde{\mathcal{O}}$. Then, user 3, as a class-3 user, neither adopts nor recommends. At the beginning of time slot 2, user 1 arrives and recommends to user 2. Then in time slot 3, user 2 arrives and recommends to user 3. Finally in time slot 4, user 3 revisits the item because of the recommendation. After that the diffusion terminates as no users will adopt or recommend the item anymore.

Remark. Note that our model is a generalization of the *"Independent Cascade model"* used in the influence maximization problem [14], where users always recommend an item once they are informed of it. In this case, whether a user can receive information (or "be influenced") of item i only depends on the weights of the edges $[w_{mn}{:}m, n{\in}\mathcal{N}]$, given the set of users who receive information from other sources. Another information diffusion model is the *"Linear Threshold model"* [14], where a user will recommend if a summantion of friends' influences exceeds a threshold. We conduct experiments in Section 6 to compare these models.

To characterize a firm's uncertainty about the market, we model a user n's valuation v_{ni}, the social value $\tilde{v}_{ni}$ and recommendation cost $\tilde{c}_n$ as random variables. We assume $(v_{ni}, \tilde{v}_{ni}{-}\tilde{c}_n)$ are independent across different n and i. Moreover, the parameters $(v_{ni}, \tilde{v}_{ni}{-}\tilde{c}_n)$ follow a probability distribution $\mathcal{D}_{ni}$. From the perspective of a firm, a user n in state S_{ni} will recommend (or adopt) an item i with certain

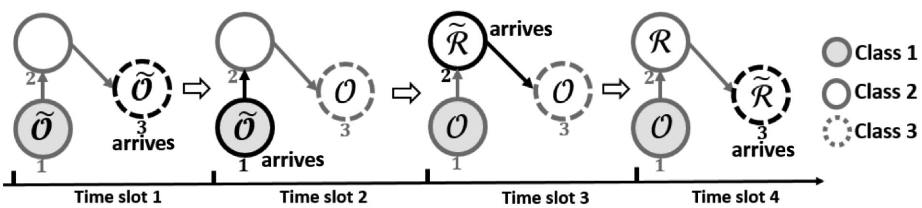


Fig. 9. Diffusion of an item's information. (User n has class n in Fig. 8, $n{=}1,2,3$).

probability. We define a user n's probability to adopt and recommend the item i to be $a_{ni}(S_{ni})$ and $q_{ni}(S_{ni})$ respectively, where $S_{ni}\in\{\mathcal{O},\mathcal{R}\}$. Note that $a_{ni}(S_{ni})$ are $q_{ni}(S_{ni})$ are personalized and item-specific. For example, when users maximize their estimated utilities, the probabilities are

$$\begin{aligned}a_{ni}(S_{ni})&=\mathbb{P}_{(v_{ni},\tilde{v}_{ni}-\tilde{c}_n)\sim\mathcal{D}_{ni}}[\hat{u}_{ni}(S_{ni})+g_{ni}>0 \text{ or } \hat{u}_{ni}(S_{ni})>0],\\ q_{ni}(S_{ni})&=\mathbb{P}_{(v_{ni},\tilde{v}_{ni}-\tilde{c}_n)\sim\mathcal{D}_{ni}}[\hat{u}_{ni}(S_{ni})+g_{ni}>0 \text{ and } g_{ni}>0].\end{aligned} \tag{7}$$

We point out that the reward and price affects both a user's recommendation probability and his adoption probability. This is because the price determines the *estimated utility from adoption*, i.e., $\hat{u}_{ni}(S_{ni})$, and the reward determines the *utility gain to recommend*, i.e., g_{ni}. As shown in (7), $\hat{u}_{ni}(S_{ni})$ and g_{ni} jointly affect the recommendation and adoption probabilities. In our model, a user will jointly decide whether to recommend or adopt an item, as users will take the reward into account when deciding whether to buy an item.

As shown in Fig. 8, whether a user n (in any class) eventually adopts or recommends an item i is determined only by her state S_{ni} when the diffusion terminates. Recall that we consider the case where the firm has the same price p_i and reward r_i for all users. We can then derive the demand (i.e., expected number of adoptions) of item i as

$$D_i(p_i,r_i)\triangleq\sum_{n\in\mathcal{N}}\left(\sum_{S\in\{\mathcal{O},\mathcal{R}\}}\mathbb{P}\big[S_{ni}^{\mathrm{Ter}}=S\big]a_{ni}(S)\right),$$

where S_{ni}^{Ter} denotes the final state of a user when the diffusion terminates. Similarly, we derive the expected number of users who make recommendations of item i as

$$R_i(p_i,r_i)\triangleq\sum_{n\in\mathcal{N}}\left(\sum_{S\in\{\mathcal{O},\mathcal{R}\}}\mathbb{P}\big[S_{ni}^{\mathrm{Ter}}=S\big]q_{ni}(S)\right). \tag{8}$$

Then item i's *"net profit"* is $D_i(p_i,r_i)\cdot(p_i-c_i)-R_i(p_i,r_i)\cdot r_i$.

We consider the problem of optimal rewarding scheme when the firm has uniform price and reward for all users.

Problem 1 (uniform). *The optimal of finding the optimal uniform rewarding scheme problem is*

$$\underset{p_i\geq 0,r_i\geq 0}{maximize}\, P_i(p_i,r_i)\triangleq\{D_i(p_i,r_i)\cdot(p_i-c_i)-R_i(p_i,r_i)\cdot r_i\}.$$

Even through we currently only allow a firm to set a uniform price and rewardfor all users, the above problem is challenging to solve. Increasing the reward r_i improves the demand, but also the firm will incur higher expenses. Decreasing the price p_i improves the demand, but also hurts the firm's profit gain. The problem becomes more difficult when the price and reward jointly affect the profit.

4 Model Analysis

In this section, we first characterize the demand. Then we characterize the optimal reward and the optimal price. These analyses identify key factors that affect the firm's rewarding strategies and customers' utilities, which guides us to do more numerical simulations in Section 6.

Characterizing the Demand. To make the analysis tractable, we consider the case in which $\mathcal{D}_{ni},\tau_n,\tilde{\tau}_n$ are identical across different users n, and $\tau_n=\tilde{\tau}_n$. Namely, we assume all users' utilities have identical distributions. This assumption corresponds to the setting where all types of users are randomly distributed in the social network. This setting is common in theoretical analysis of rewarded social recommendation such as [5] and [3]. Therefore, we can abbreviate the $a_{ni}(S_{ni})$ and $q_{ni}(S_{ni})$ as a_i and q_i respectively. The following lemma characterizes the demand when δ_i is small.

Lemma 1. *Suppose $\mathcal{G}=(\mathcal{N},\mathbf{W})$ degenerates to an unweighted and undirected graph, i.e., $w_{mn}\in\{0,1\}$ and $w_{mn}=w_{nm}$ for $\forall m,n\in\mathcal{N}$. If the degree of nodes are bounded, then there is a critical value q_i^c such that*

1) $q_i<q_i^c \Rightarrow \lim_{|\mathcal{N}|\to\infty,\delta_i\to 0}\frac{D_i}{|\mathcal{N}|}=0$,
2) $q_i>q_i^c \Rightarrow \lim_{|\mathcal{N}|\to\infty,\delta_i\to 0}\frac{D_i}{|\mathcal{N}|}>0$.

Furthermore, if $\mathcal{G}$ is a random graph [4] with degree distribution $\{\rho_k\}_{k=0}^{+\infty}$, then $q_i^c=\big(\sum_{k=1}^{+\infty}k\rho_k\big)/\big(\sum_{k=2}^{+\infty}k(k-1)\rho_k\big)$.

Proof Sketch. We map the diffusion of recommendations to a percolation process [4], [5]., where the recommendation probability maps to *"occupation probability"*. Due to page limit, detailed proofs are in our supplement, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TKDE.2020.3038930. □

Lemma 1 states that only if the recommendation probability is higher than a critical value q_i^c, then an infinitesimal fraction ($\delta_i\to 0$) of users who are informed from other sources (e.g., traditional advertisements) can boost a positive fraction of user to adopt the item. This implies that a firm can offer reward r_i so that the recommendation probability q_i will exceed q_i^c. For large q_i, we have the following lemma.

Lemma 2. *Suppose $\mathcal{G}$ is a random graph [4] with degree distribution $\{\rho_k\}_{k=0}^{\infty}$, then $\lim_{q_i\to 1}\frac{dD_i}{dq_i|\mathcal{N}|}=0$ provided that $\rho_1=0$.*

In Lemma 2, recall that D_i is the demand of item i, q_i is users' recommendation probability. Then $dD_i/(dq_i|\mathcal{N}|)$ represents the increment of demand D_i w.r.t. the increment of q_i, normalized by the number of users $|\mathcal{N}|$ in the social network. Lemma 2 states that if the recommendation probability is high enough, then further increasing the recommendation probability only increases the demand marginally.

Simulations on Yelp's graph (Fig. 10) validate our previous analytical results, and show that the demand is small when the recommendation probability is smaller than a critical value 0.02. Also, the demand increases slowly when recommendation probability is high enough.

Characterizing Optimal Reward and Price. The following theorem characterizes the optimal reward r_i^*. Note that only when the optimal reward r_i^* is positive, the firm should adopt the scheme to provide rewards to recommenders.

Theorem 1. *If the parameters satisfy the following condition: $\left.\left[\frac{dq_i}{dr_i}\frac{q_i dD_i}{D_i dq_i}(p_i-c_i)-\frac{q_i^2}{a_i}\right]\right|_{r_i=0}>0$, then $r_i^*>0$.*

Proof Sketch. We express derivatives of the profit $P_i(p_i,r_i)$ w.r.t. p_i and r_i, then we can get the optimal p_i^* and r_i^*. □

Theorem 1 states a sufficient condition under which a firm should adopt the rewarding scheme; This theorem uncovers

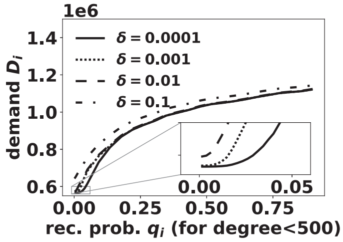


Fig. 10. [Yelp] Recommendation prob. q_i and demand D_i.

key factors that determine a firm's profitability: the recommendation probability q_i, the adoption probability a_i, the demand D_i etc. This guides us to study the impacts of these factors by numerical simulations in Section 6.

To study how the social recommendation scheme affects the price of an item, we compare a firm's optimal reward with and without social recommendations. Recall that the optimal price under the social recommendation scheme is p_i^*. We denote the optimal price as $p_i^\emptyset$ when users do not have OSN, and we denote the optimal price as p_i^v when users do not receive rewards and can only do voluntary recommendations. We also denote the optimal profit of firm i in the above three settings as P_i^*, $P_i^\emptyset$ and P_i^v respectively. We summarize the three settings in Table 6.

Theorem 2. *Suppose $\mathcal{G}$ is a random graph [4] with degree distribution $\{\rho_k\}_{k=0}^{\infty}$, γ_i=1 and c_i=0. Then we have: (1) $P_i^* \geq P_i^v \geq P_i^\emptyset$; (2) $p_i^\emptyset \geq p_i^v$; (3) If $(v_{ni}, \tilde{v}_{ni} - \tilde{c}_n)$ follows a two-dimensional uniform distribution, then $p_i^* \geq p_i^\emptyset \geq p_i^v$;*

First, a firm can have the highest profit if he can provide incentives for recommenders. Second, voluntary recommendations will result in a lower price thus improve customers' utilities. Third, incentivized recommendations could hurt the utilities of non-recommenders who suffer from a higher price but do not get rewards.

5 Parameter Inference & Model Validation

In this section, we infer model parameters from social recommendation data of a set of items. We also evaluate the accuracy of our inferred model on the Weixin's dataset.

5.1 Parameter Inference

Data Model. Consider a social recommendation outcome dataset, which is item-centric, i.e., for each item $i \in \mathcal{I}$, the data contains: (1) the price p_i (or discount $-\Delta p_i$) and reward r_i; (2) whether a user n recommends it, i.e., the value of $R_{ni} \in \{0,1\}$; (3) whether a user n receives recommendations of item i, indicated by $\tilde{R}_{ni} \in \{0,1\}$ ($\tilde{R}_{ni}$=1 means user n receives and $\tilde{R}_{ni}$=0 means not). Furthermore, we know the social network graph except for the weights **W** on edges.

Inferring Edge Weights. Recall that the weight w_{mn} represents the probability that user n can receive a recommendation from user m. When user m has at least one recommendation (i.e., $\sum_{i\in\mathcal{I}} R_{mi} \geq 1$), we infer w_{mn} as

$$w_{mn} = \frac{(\text{num. of recommendations } n \text{ receives from } m)}{\left(\sum_{i\in\mathcal{I}} R_{mi}\right)}.$$

When the edge (m,n) exists, but user n has no recommendations in the data, we set w_{mn} to be the system average

TABLE 6
The Optimal Prices and Profits in Three Settings

Setting	OSN?	reward?	opt. price	opt. reward	opt. profit
Incentivized recommendation	✓	✓	p_i^*	r_i^*	P_i^*
No recommendation	×	N/A	$p_i^\emptyset$	N/A	$P_i^\emptyset$
Voluntary recommendation	✓	×	p_i^v	0	P_i^v

$$\overline{w} \triangleq \sum_{m,n\in\mathcal{N}} w_{mn} \mathbf{1}_{\{\sum_{i\in\mathcal{I}} R_{ni} \geq 1\}} \Big/ \sum_{m,n\in\mathcal{N}} \mathbf{1}_{\{\sum_{i\in\mathcal{I}} R_{ni} \geq 1\}}. \tag{9}$$

Parameters for an Item's Diffusion. In our model, the parameters $\delta_i q_{ni}(\mathcal{O})/q_{ni}(\mathcal{R})$ and $\{q_{ni}(\mathcal{R})\}_{n\in\mathcal{N}}$ fully describe a diffusion process, as shown by the following Theorem.

Theorem 3. *The diffusion process is fully described by a tuple of parameters $\langle \delta_i, \{q_{ni}(\mathcal{O})\}_{n\in\mathcal{N}}, \{q_{ni}(\mathcal{R})\}_{n\in\mathcal{N}} \rangle$. Then, the probability for a user n to recommend an item i, i.e., $\mathbb{P}[R_{ni}=1]$, is the same under the following two tuples of parameters*

- $\langle \delta_i, \{q_{ni}(\mathcal{O})\}_{n\in\mathcal{N}}, \{q_{ni}(\mathcal{R})\}_{n\in\mathcal{N}} \rangle$
- $\langle \delta_i q_{ni}(\mathcal{O})/q_{ni}(\mathcal{R}), \{q_{ni}(\mathcal{R})\}_{n\in\mathcal{N}}, \{q_{ni}(\mathcal{R})\}_{n\in\mathcal{N}} \rangle$

provided that $q_{ni}(\mathcal{R})/q_{ni}(\mathcal{O})$ is a constant for every user n.

Recall that user n recommends item i with probability $q_{ni}(\mathcal{O})$ when he knows item i from other information source (in the state $\mathcal{O}$), and he recommends item i with probability $q_{ni}(\mathcal{R})$ when he knows item i from friends' recommendations (in the state $\mathcal{R}$). The reason why the two sets of parameters are equivalent in Theorem 3 is as follows. When user n does not receive any recommendation, he recommends item i with the same probability $\delta_i q_{ni}(\mathcal{O})$ under both sets of parameters. When user n receives recommendations, he recommends item i with the same probability $q_{ni}(\mathcal{R})$, under both sets of parameters. In short, no matter whether a user receives recommendations or not, he recommends an item with the same probability under both sets of parameters.

Then, one can derive that $\delta_i \frac{q_{ni}(\mathcal{O})}{q_{ni}(\mathcal{R})} = \frac{\mathbb{P}[R_{ni}=1|\tilde{R}_{ni}=0]}{\mathbb{P}[R_{ni}=1|\tilde{R}_{ni}=1]}$, so we use the empirical value of $\frac{\mathbb{P}[R_{ni}=1|\tilde{R}_{ni}=0]}{\mathbb{P}[R_{ni}=1|\tilde{R}_{ni}=1]}$ to infer $\delta_i \frac{q_{ni}(\mathcal{O})}{q_{ni}(\mathcal{R})}$

$$\delta_i \frac{q_{ni}(\mathcal{O})}{q_{ni}(\mathcal{R})} = \frac{\sum_{n\in\mathcal{N}} R_{ni}(1-\tilde{R}_{ni}) / \sum_{n\in\mathcal{N}} (1-\tilde{R}_{ni})}{\sum_{n\in\mathcal{N}} R_{ni} \times \tilde{R}_{ni} / \sum_{n\in\mathcal{N}} \tilde{R}_{ni}}. \tag{10}$$

Note that we do not observe whether a user knows the item from other information sources. Hence, we do not know δ_i. What we observe from data is users' recommendation behavior. By Theorem 3 and Equation (10), we do not need to infer δ_i, but can use $\delta_i q_{ni}(\mathcal{O})/q_{ni}(\mathcal{R})$ instead, which are able to be inferred from data.

To infer the recommendation probabilities $\{q_{ni}(\mathcal{R})\}_{n\in\mathcal{N}}$ of each users, we categorize users into types, and estimate $\{q_{ni}(\mathcal{R})\}_{n\in\mathcal{N}}$ for each type. For example, we can categorize the users by their degree, or demographic features, etc. Let us denote the type of user n as $T_n \in \mathcal{T}$ where $\mathcal{T}$ is the set of all types. Then, the probability for a type-T user to recommend an item i in state $\mathcal{R}$ is inferred as

$$q_{T,i} = \left(\sum_{n,T_n=T} R_{ni} \times \tilde{R}_{ni} \right) \Big/ \left(\sum_{n,T_n=T} \tilde{R}_{ni} \right).$$

TABLE 7
MAPE for Predicting the Number of Recommenders (Weixin, More Than 50 Items)

	IC Model (Item-independent)	Our model (One type)	Our model (four types)	Our model (4K+ types)
MAPE	68.8% (at best)	13.9%	**11.7 %**	11.6%

Our model can accurately predict the outcomes of diffusion processes even though the diffusion processes are random. Our model (with 4 types) can reduce the error of IC model by more than 80 percent.

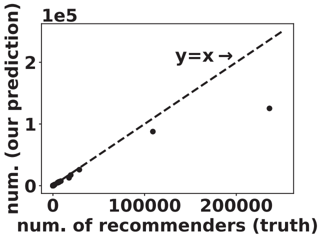


Fig. 11. [Weixin] Predict the fraction of recommenders with four types of users.

Parameters for Users' Recommendation Behavior. Inferring users' behaviors when firms have different rewarding schemes is a counterfactual task, since the available data is generated from firms' current reward schemes. As stated in our model, various factors affect a user n's recommending probability on an item i. These factors include the price p_i, the reward r_i, user type T_n, and item's features that are represented by a vector $\boldsymbol{x}_i$. Since we are interested in the impact of price and reward, users' type T_n and item's features $\boldsymbol{x}_i$ are *"confounding variables"* (a term in causal inference [20]). We assume that $q_{T,i}$ obeys a linear form

$$q_{T,i} \simeq C_T(\theta_0 + \theta_1 p_i + \theta_2 r_i + \boldsymbol{\theta}_X \boldsymbol{x}_i), \tag{11}$$

where C_T is a constant for type T and $\boldsymbol{\theta}_X$ is a vector of parameters for item's features. For notational convenience, we denote $\boldsymbol{\theta} \triangleq (\theta_0, \theta_1, \theta_2, \boldsymbol{\theta}_X)$.

Now, we describe our inferring procedure. First, we calculate $\widehat{q}_i \triangleq \left(\sum_{n\in\mathcal{N}} R_{ni} \times \tilde{R}_{ni}\right) / \left(\sum_{n\in\mathcal{N}} \tilde{R}_{ni}\right)$ which is the frequency for a user to recommend item i after receiving recommendations. Second, we fit a linear function $L_q(p, r; \boldsymbol{\theta}) = \theta_0 + \theta_1 p + \theta_2 r + \boldsymbol{\theta}_X \boldsymbol{x}_i$ via ordinary least squares. The inferred parameters are $\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{i\in\mathcal{I}} (\widehat{q}_i - L_q(p_i, r_i; \boldsymbol{\theta}))^2$. Third, we infer different types of users' recommendation probability. Suppose a type-T user's recommendation probability is $q'_{T,i}(p'_i, r'_i)$ when the price and reward change to p'_i and r'_i. Note that the recommendation probability of a user in type T is proportional to C_T according to our assumption (11), then

$$q'_{T,i}(p'_i, r'_i) = q_{T,i} \times L_q(p'_i, r'_i; \widehat{\boldsymbol{\theta}}) / L_q(p_i, r_i; \widehat{\boldsymbol{\theta}}). \tag{12}$$

Users' adoption probabilities are inferred similarly. Since items' price is the major factor that affect users' adoption behaviors, we use a linear function $L_a(p; \widehat{\boldsymbol{\beta}})$ to fit $\{\widehat{q}_i\}_{i\in\mathcal{I}}$. Under price p'_i, a type-T user adopts item i with probability

$$a'_{T,i}(p'_i) = a_{T,i} \times L_a(p'_i; \widehat{\boldsymbol{\beta}}) / L_a(p_i; \widehat{\boldsymbol{\beta}}). \tag{13}$$

If we know a firm's discount instead of price in the data, one can replace p_i with Δp_i in the above inference method.

5.2 Model Validation

We validate our model via designing the following two prediction tasks based on Weixin's Data.

Predicting an Item's Diffusion on OSN. The number of recommenders reflects how an item's information diffuses in OSN. The number of recommenders is also *nearly linear* in the number of recommendation receivers (Fig. 4a). To evaluate the models, we compare the ground truth on the number of recommenders and the models' predictions. In the evaluation, we use a random graph [4] with the same degree distribution as Weixin's social network. Let $N_i^{(R)}$ be the true number of recommenders, and $\widehat{N}_i^{(R)}$ be a model's prediction. We use the *Mean Absolute Percentage Error* (MAPE) as our evaluation metric

$$MAPE \triangleq \left(\sum_{i\in\mathcal{I}} |N_i^{(R)} - \widehat{N}_i^{(R)}| / N_i^{(R)}\right) / |\mathcal{I}|.$$

Table 7 compares the MAPE's of the independent cascade (IC) model and our model (with different number of types). For our model, we classify the type of nodes according to their degrees because of the observation in Fig. 5b First, we treat all nodes as a single type. Second, we group nodes whose degrees are in regions [0,20), [20,60), [60,200), [200,∞) into four types. Third, we group nodes with the same degree into one type, and we get more than 4,000 types. For all these predictions, we use a uniform weight $\overline{w}$ on each edge which is defined in (9). Since we do not train the model with labeled data, we do not split the dataset for training and testing.

The "Independent Cascade model"[13], [14] serves as the baseline. For the IC model, users' recommendation probability q=1, and the number of recommenders only depends on the initial users who know the item from other information sources. In the evaluation, we allow the independent cascade model to tune the uniform weight w on edges, so we get the lowest MAPE that the IC model can achieve, i.e., 68.8 percent. As pointed out by authors of the paper [13], IC model ignores users' different reactions for different items, hence it cannot capture different items' diffusion outcomes. That is why our model significantly outperforms the IC model.

Fig. 11 shows that our predictions on the fraction of recommenders matches well with the ground truth. In Fig. 4b, we plot the critical value q_i^c=0.035 predicted by our model (dotted line), which matches our previous observation.

We compare the prediction accuracy of our model and that of the Linear Threshold (LT) model. For fair comparision, we allow different items to have different probability configurations in the LT model. Fig. 12 shows the prediction performance of the LT model. The MAPE for the LT model is 176.79 percent, which is much higher than the MAPE 11.7 percent of our model. In particular, when the true number of recommenders is large, the predictions of LT model has a large deviation from the true value. The reason might be that the critical value of recommendation probability in our model does not apply to the LT model. Therefore, the two models have different predictions on items whose recommendation probability is close to the critical value of our model. The experiment details can be found in our supplementary material, available online.

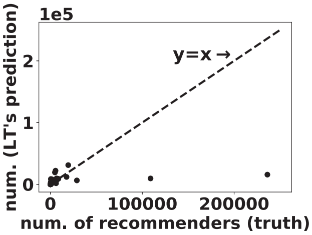


Fig. 12. [Weixin] Predict the fraction of recommenders via linear threshold model.

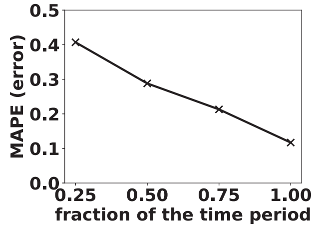


Fig. 13. Predict the diffusion using early-stage statistics.

Moreover, Fig. 13 shows how well our model can use early-stage statistics to predict the total number of recommenders in the whole time period. In Fig. 13, a fraction of "0.25" means that we only use the data in the first 1/4 of the time period. We see that the prediction error decreases as we have data in a longer period of time. Moreover, when we have data in the first half of the time period, our model can achieve a MAPE as low as 30 percent.

Predicting Users' Recommendation Behavior. We collected recommendation behaviors associated with 27 items from four firms, including the nine items in Table 1. The items of each firm have similar characteristics. In this data, users receive a total number 7,994,844 of recommendations.

The objective is to predict the user population's recommendation probability. For each firm $j = 1, 2, 3, 4$, we use an indicator function $\mathbf{1}_{F_j}$ to denote whether the item belongs to the jth company. Our regression model is written as

$$q = \theta_0 + \theta_1 \Delta p + \theta_2 r + \left(\sum_{j \in \{1,2,3,4\}} \mathbf{1}_{F_j} \theta_{j+2} \right) + \theta_7 \times \text{click rate}.$$

Here, $(\mathbf{1}_{F_1}, \mathbf{1}_{F_2}, \mathbf{1}_{F_3}, \mathbf{1}_{F_4}, \text{click rate})$ are an item's features $\boldsymbol{x}_i$ in Equation (11). In the 7,994,844 recommendations received by users, we randomly divide 80 percent of them as the training data and use the remaining 20 percent as the testing data. We only use the training data to train the linear regression model. The mean squared error of our trained model on the testing data is 1.13×10^{-4}, which means that for any item the relative error on recommendation probability is at most 5.96 percent. If we remove the factor of the "reward" r in the above regression model, the mean squared error is 1.42×10^{-4}, which is 26 percent higher than the previous error of the full model. It shows that the reward plays an important role in predicting users' recommendation probability. We point out that one can hardly validate counterfactual predictions using offline data, so a firm needs to learn users' behaviors in an online setting (which we will introduce in Section 7).

Predicting Individualized Metrics. Previously, we predict users' recommendation behaviors at a population level. Here, we investigate whether our data can predict each individual user' clicking behavior.

We collect 40 user features and seven item features, and use the XGBoost 0-1 classifier [6] to predict whether a recipient of recommendation will click the recommended item. The Rooted Mean Squared Error is 0.178, and the Area Under Curve (AUC) is 0.735. It indicates that the users' features and items' characteristics can to some degree predict whether a user will click an item. Detailed results can be found in the supplementary material, available online.

6 Simulations With Inferred Parameters

With our model and parameters inferred from Weixin and Yelp datasets, we simulate social recommendations. Then, we uncover important insights on when social recommendations can (or cannot) improve firms' profit and users' utilities. Here, the cost c_i is set to 0 so that the parameter p_i stands for $p_i - c_i$.

6.1 Algorithms for Simulation

Recall that it is NP-hard to compute the expected number of receivers of information in general graphs [7]. To address this challenge, we design efficient Monte-Carlo algorithms.

Note that the time complexity to collect one sample by simulating the diffusion of recommendation is $\Omega(|\mathcal{N}|)$, since it is possible that the recommendation diffuses across the whole network. Worse yet, we are interested in many parameter settings and for each parameter setting, we need to collect enough samples in the Monte-Carlo simulation. To address these challenges, we design Algorithm 1 that sequentially generates the samples, so that the time complexity to get one sample is $O(1)$. The idea to sequentially generate samples comes from Newman's book [18]. From all type-T nodes, we select one node v uniformly randomly to be the representative node. To generate the samples, we repeatedly select a new node u who is willing to recommend after receiving information, and put it into the set $\mathcal{S}_R$, i.e., $\mathcal{S}_R \leftarrow \mathcal{S}_R \cup \{u\}$. For a set of users $\mathcal{S}_R$ who are willing to recommend, the reverse reachable set $RRSet(v, \mathcal{S}_R)$[23] is defined as the set of users, where if any user in the set $\mathcal{S}_R$ is informed by other information, the user v will be informed from recommendation. When $\mathcal{S}_R$ has cardinality n, we collect a sample s_n which records the number of nodes in $RRSet$, and the number of selected users in each type $|\mathcal{S}_R \cap \mathcal{N}_T|$, $\forall T \in \mathcal{T}$, where $\mathcal{N}_T$ denotes a set of type-T users.

Using the samples generated by Algorithm 1, Algorithm 2 shows how we estimate the expected probability for a user to be informed of an item via the "importance sampling" technique. We can derive that the probability to get a sample $s = \big(|RRSet(v, \mathcal{S}_R)|, \{|\mathcal{S}_R \cap \mathcal{N}_T|\}_{T \in \mathcal{T}}\big)$ is

$$w(s) = \prod_{T \in \mathcal{T}} \left[\binom{|\mathcal{N}_T|}{|\mathcal{S}_R \cap \mathcal{N}_T|} q_{T,i}(\mathcal{R})^{|\mathcal{S}_R \cap \mathcal{N}_T|} (1 - q_{T,i}(\mathcal{R}))^{|\mathcal{N}_T| - |\mathcal{S}_R \cap \mathcal{N}_T|} \right].$$

Given the reverse reachable set $RRSet(v, \mathcal{S}_R)$, the probability for the randomly selected node v to be informed from friends' recommendation is $p(s) = 1 - (1 - \delta)^{|RRSet(v, \mathcal{S}_R)|}$, since each user in the reverse reachable set is independently informed from other information with probability δ.

TABLE 8
Running Time to Estimate a Firm's Profit for Different Parameters (Monte-Carlo Simulation Using 2000 Samples on Yelp's Graph)

Method	Average time	(Min, Max) time
Agent-based	2970.6s	(2561.8s, 3365.7s)
Pre-computed RR set (ours)	**0.083s**	**(0.016s, 0.55s)**

Algorithm 1. [Pre-Compute Phase] Generating Samples for a Type T Node

Function get_sequence_samples:
  $v \leftarrow$ drawn uniformly randomly from all type-T nodes
  $\mathcal{S}_R \leftarrow \emptyset$
  for $n = 1$ *to* $|\mathcal{N}|$ **do**
    $u \leftarrow$ a node selected randomly in $\mathcal{N} - \mathcal{S}_R$
    $\mathcal{S}_R \leftarrow \mathcal{S}_R \cup \{u\}$
    // $\{RRSet(v, \mathcal{S}_R)$ is Reverse Reachable Set
    $s_n \leftarrow (|RRSet(v, \mathcal{S}_R)|, (|\mathcal{S}_R \cap \mathcal{N}_T|)_{T\in\mathcal{T}})$
  return $(s_n)_{n=1}^{|\mathcal{N}|}$ // generating the samples
for $k = 1$ *to* K **do**
  $(s_{k,n})_{n=1}^{|\mathcal{N}|} \leftarrow$ **get_sequence_samples**()

Algorithm 2. [Evaluation Phase] Estimate the Prob. That a Type-T Node is in State $\mathcal{R}$ After Diffusion

Function get_sequence_samples(δ_i, $\{q_{T,i}(\mathcal{R})\}_{T\in\mathcal{T}}$):
  $\hat{P}_{T,i} \leftarrow \frac{\sum_{k\leq K, n\leq|\mathcal{N}|} w(s_{k,n})p(s_{k,n})}{\sum_{k\leq K, n\leq|\mathcal{N}|} w(s_{k,n})}$
  return $\hat{P}_{T,i}$

Using Algorithm 2, we can reuse the pre-computed $RRset$ to do estimation for any parameter settings. Table 8 shows that the running time of our algorithm with pre-computed $RRset$ is less than 1/6,000 of the running time of the agent-based simulation.

For readers to better understand the above two algorithms, we use an example to walk through the two algorithms in Appendix B.1 in the supplementary material, available online.

For our simulation algorithms, we coduct scalability analysis on both the pre-computing phase and the evaluation phase. We randomly choose a certain fraction of users in Yelp's social network and only keep the links among these selected users. Fig. 14 plots the running time of 10,000 times evaluation. We observe that the evaluation time increases linearly as the number of users increases in the social network. This is because the size of the intermediate file generated by the pre-computing phase is linear w.r.t the number of users, and we need to went through the intermediate file for evaluation. In addition, the running time of pre-computing phase increases in a $O(n^2)$ order where n is the number of users. This is because the pre-computing algorithm needs to go over the edges in the social network and the number of edges scales in a $O(n^2)$ order in our setting of graph generation.

Theorem 4. $\mathbb{E}[\hat{P}_{T,i}] = \mathbb{P}[S_{ni}^{\mathrm{Ter}} = \mathcal{R}|T_n{=}T]$. *Namely,* $\hat{P}_{T,i}$ *is an unbiased estimator of the probability that a user is in state* $\mathcal{R}$.

Proof. This is importance sampling with inverse propensity score. Especially, the propensity score is accurate. By importance sampling's theory [22], our estimator is unbiased. □

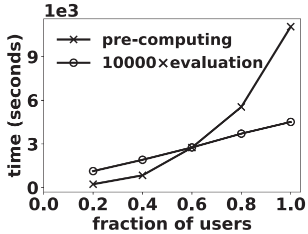


Fig. 14. Scalability analysis of our algorithms.

6.2 Simulating Weixin's Social Recommendations

We use the parameters inferred from Weixin dataset as the input of our model and do simulations, where one firm has one representative item. Fig. 21 shows that social recommendations can improve firms' profits for 9 out of 15 items, by 7%~71%. In Fig. 22, the items are ranked by their click rates, where the right-most items have the lowest click rates. We observe that when an item's optimal price with the current reward is higher than the optimal price without reward, the item has a low click rate. In other words, our model explains the observation in Fig. 6 that "users' low click rates are associated with rewards" as follows: if a firm offers rewards, then the firm will increase the price which hurts users' utilities, hence users have low click rates. Fig. 19 shows that when firms do not increase prices (and users' utilities are not hurt), social recommendations can improve 4 firms' profits by 7%~21%.

Lessons Learned. Firms will increase the price after providing reward. This explains the association between items' reward and users' low click rates. When firms do not increase price, social recommendations can improve both firms' profit and users' utilities.

6.3 Simulating Yelp's Social Recommendations

We input the Yelp's graph into our model to do simulations, and further uncover the conditions when social recommendation can (or cannot) improve firms' profits and users' utilities. The detailed settings are in our supplement file, available online.

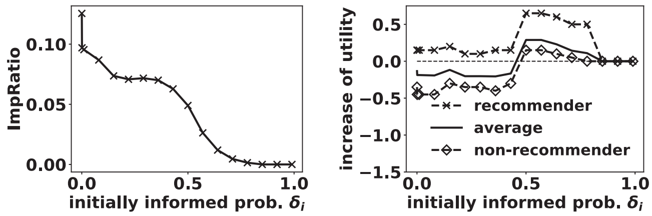
6.3.1 Firms' Profitability

To quantify the improvement of a firm's profit by rewarding recommenders, we compare the profits with/without reward, and define improvement ratio as

$$\mathrm{ImpRatio} \triangleq \frac{\max_{p_i, r_i \in \{0.05m\}_{m=0}^{40}} P_i(p_i, r_i)}{\max_{p_i \in \{0.05m\}_{m=0}^{40}} P_i(p_i, 0)} - 1.$$

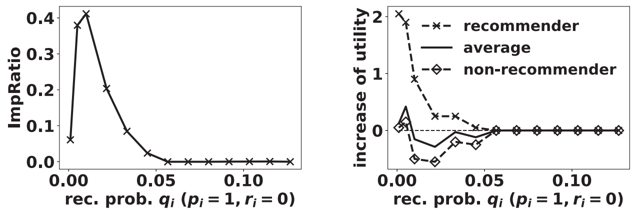
• *Information from other source* δ_i. Fig. 15a shows that the improvement ratio decreases as δ_i increases. This implies that incentivized recommendations are benefitial only for firms whose items are not well-known by the public.

• *Recommendation probability* q_i. In Fig. 16a, as the recommendation probability increases, we observe that the improvement ratio first increases and then decreases. Moreover, when the initial recommendation probability is close to the critical value $q_i^c = 0.02$ (which matches the critical value in Fig. 10), the improvement ratio reaches the highest value. This is because when users' recommending probability is close to the



(a) **Firm's profit** (b) **Users' utility**

Fig. 15. Impact of prob. to be informed from other sources.



(a) **Firm's profit** (b) **Users' utility**

Fig. 16. Impact of users' recommending probability.

critical value, a small increment of recommending probability driven by rewards can cause a significant improvement of demand. But when users' recommendation probability is originally high (>0.05) without reward, a firm should *not* offer rewards for social recommendations, as the ImpRatio is 0.

Lessons Learned for the Firms. Rewarding recommenders can improve a firm's profit when the item is not well-known, and users' recommending probability is near a critical value for outbreak of diffusion. Otherwise, firms will not be profitable by using social recommendations.

6.3.2 Users' Utility

Since a firm's rewarding scheme does not change its item's quality, the item's price is the only factor that affects users' utilities. The *"baseline price"* is a firm's optimal price without rewards, i.e., $p_i^v \triangleq \arg\max_{p_i} P_i(p_i, r_i{=}0)$. Under the rewarding scheme, the firm sets price to p_i^* and sets reward to r_i^*. Then, a recommender needs to pay $p_i^* - r_i^*$. Compared to the baseline price, his *"increase of utility"* is $p_i^v{-}(p_i^*{-}r_i^*)$. Similarly, a non-recommeder has an increase of utility $p_i^v - p_i^*$. We also consider the *"average increase of utility"* of both recommenders and non-recommenders.

• *Information from other sources* δ_i. Fig. 15b shows that when the proportion $\delta_i > 0.5$ of users know the item from other sources, social recommendations improve customers' average utility. This is because a firm will not increase price to lose the large portion of users who know the item from other sources. Also, firms' rewards improve the recommenders' utilities.

• *Recommendation probability* q_i. We observe in Fig. 16b that when users' recommending probability is very small ($q_i < 0.01$) without rewards, social recommendation improves users' average utility. This is because only by both decreasing the price and offering reward, a firm can let the recommendation probability exceed its critical value. However, as users' recommendation probability becomes higher ($q_i{\in}[0.01, 0.08]$), social recommendation hurts the non-recommenders' utilities.

• *Perceivability* γ_i. In Fig. 17, we observe that social recommendations hurt the utilities of non-recommenders when

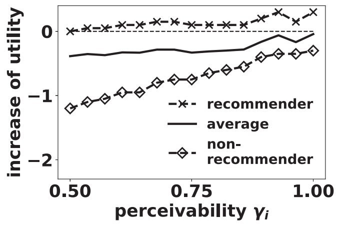


Fig. 17. Perceivability γ_i and users' utilities ($\tau{=}0, \tilde{\tau}{=}{-}0.5$).

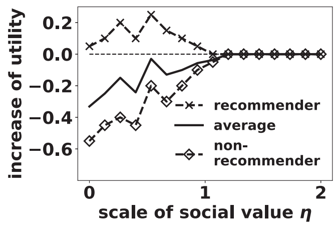


Fig. 18. Social value's scale and users' utilities, $\tilde{v}_{ni}{=}\eta(1{-}p_i)$.

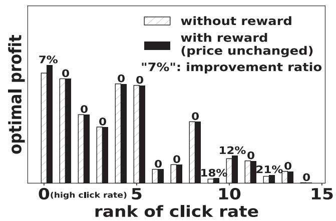


Fig. 19. Profit improvement with items' prices unchanged.

the perceivability is low. The reason is that when an item has lower perceivability, users rely more on their friends to make decisions. Hence, firms choose to offer high rewards to recommenders and increase prices.

• *Social value.* We set social value as $\tilde{v}_{ni} = \eta(1 - p_i), \eta \geq 0$. A larger η indicates that users care more about their friends' utilities. In OSNs whose users have weak-ties with friends (e.g. Twitter), η is small. From Fig. 18, one can observe that the non-recommenders' utilities are heavily hurt when η is small. This is because when users do not care about friends' utilities, firms will offer high reward to recommenders and increase prices.

Lessons Learned for the OSN Platforms. Social recommendation benefits both firms and users when an item is known by a large portion of users from other sources, or an the recommending probability of an item without rewards is low. The OSN platform can encourage social recommendations for such items. However, the platform should prohibit social recommendations for items whose quality cannot be perceived before purchasing. Moreover, social recommendation will heavily hurt non-recommenders' utilities in "weak-tie" OSNs (e.g., Twitter).

7 Extensions to Dynamic Reward and Price

We extend our model to consider the dynamic setting, where the firm can sequentially update the price and reward.

7.1 Modeling the Dynamic Settings

The models in Section 3 considers that a firm has uniform price and reward for all the users. In that case, a firm's profit

TABLE 9
Three Kinds of Heuristic Algorithms

Stable strategy	State-dependent	User-dependent
Fixed action	Reward and then stop	High degree
Random action	Q-learning	High influence

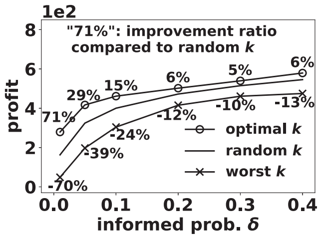


Fig. 20. Choosing k to maximize influence, $k\in\{100+300m\}_{m=0}^{9}$.

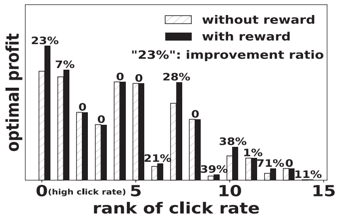


Fig. 21. Profit improvement for items in Weixin.

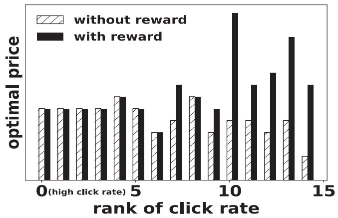


Fig. 22. Reward and the optimal price for items in Weixin.

is independent of the specific time when users make recommendations, as we will show in Appendix C of the supplementary material, available online. For dynamic rewarding strategy, one should consider the specific time when users make each recommendation. However, there are an infinite number of possible cases of the specific time of users' recommendations. We cannot cover all these cases, so we choose the following special case for further simulations. To model the order of users' arrival process, we consider a discrete time system with $t\in\mathbb{N}_+$, where an arrival of a user indicates the beginning of a new time slot. Note that the interval of a time slot is defined by the time points when users arrive. Moreover, the order of users' arrivals is the only thing we need to know to model the dynamics. Let $n^{(t)}$ denote the user who arrives in time slot t with state $S_{ni}^{(t)}\in\{\tilde{\mathcal{R}},\tilde{\mathcal{O}}\}$. Each user in states $\tilde{\mathcal{R}}$ (or $\tilde{\mathcal{O}}$) arrives independently in a Poisson process with an arrival rate $\lambda_{\tilde{\mathcal{R}}}\geq 0$ (or $\lambda_{\tilde{\mathcal{O}}}\geq 0$). Then, a user n in state $S\in\{\tilde{\mathcal{R}},\tilde{\mathcal{O}}\}$ becomes the next arriver with probability $\lambda_S/(\lambda_{\tilde{\mathcal{R}}}\times|\{n|S_{ni}^{(t)}=\tilde{\mathcal{R}}\}|+\lambda_{\tilde{\mathcal{O}}}\times|\{n|S_{ni}^{(t)}=\tilde{\mathcal{O}}\}|)$. If there are no users in state $\tilde{\mathcal{O}}$ or $\tilde{\mathcal{R}}$, then the system transits to the "Terminal" state. When user $n^{(t)}$ arrives, the firm needs to decide the reward $r_{n^{(t)}i}$ and price $p_{n^{(t)}i}$ for this user. Suppose under $p_{n^{(t)}i}$ and $r_{n^{(t)}i}$, the user's recommending and adopting probability are $q_{n^{(t)}i}$ and $a_{n^{(t)}i}$ respectively. Then the firm i's expected profit in time slot t is $R^{(t)}\triangleq(p_{n^{(t)}i}-c_i)a_{n^{(t)}i} - r_{n^{(t)}i}q_{n^{(t)}i}$.

In summary, a firm's decision process is a Markov Decision Process (MDP). The state space is $\{\emptyset,\mathcal{O},\mathcal{R},\tilde{\mathcal{O}},\tilde{\mathcal{R}}\}^{|\mathcal{N}|}\times\mathcal{N}$. In particular, state at time slot t is represented by $(\{S_{ni}^{(t)}\}_{n\in\mathcal{N}},n^{(t)})$, which corresponds to the state of each user in the OSN as well as the arriving user. If no user arrives, the state transits to the "Terminal" (or absorbing) state. Here, the firm has zero profit in "Terminal" state. A firm's action space is $\mathcal{A}\subseteq\{(p_i,r_i)|p_i\geq 0, 0\leq r_i\leq p_i\}$ which we assume to be a finite set due to real-world constraints. A firm's policy is a function $\pi(\{S_{ni}\}_{n\in\mathcal{N}},n)\in\mathcal{A}$ that prescribes a price and reward for the arriver n when users' states are $\{S_{ni}\}_{n\in\mathcal{N}}$. Formally, the problem of dynamically finding the optimal rewarding strategy is as follows:

Problem 2 (dynamic). *The problem of finding the optimal dynamic rewarding strategy is*

$$\text{maximize}_\pi \qquad \sum_{t=1}^{+\infty}\mathbb{E}[R^{(t)}].$$

As $S_{ni}^{(t)}$ has 5 possible values, thus the MDP has more than $5^{|\mathcal{N}|}$ possible states, which introduces challenges to locate the optimal policy.

7.2 Locating the Optimal Policy With Model Parameters

As a first step, we consider that the firm knows the model parameters. The baseline profit is the total maximum profit without reward. Among a firm's many possible strategies, we do not consider firms' strategies that depend on users' features (e.g., demographic features). We consider the following strategies which are also listed in Table 9:

1) *"Fixed action"*. The firm gives all users the same price and reward. This is the strategy considered in Section 6.
2) *(Stable) Randomized action.* A mixed strategy is represented as a vector $s=(s_1,\ldots,s_{|\mathcal{A}|})$ where the sum $\sum_{i=1}^{|\mathcal{A}|}s_i=1$. Under this strategy, the firm takes the ith action in $\mathcal{A}$ ($1\leq i\leq|\mathcal{A}|$) with probability s_i, for all the users.
3) *"High-degree"*. The firm only gives rewards to users whose degrees are above a certain threshold.
4) *"High-influence"*. The firm uses the algorithm in [23] to select k seed users to maximize the "influence" (defined [23]), and the firm only gives rewards to these selected users. The firm will tune the seed size k to optimize its profit.
5) *"Reward and then stop"*. The firm only gives rewards to the first-k arriving users, and the firm tunes the parameter k to optimize its profit.
6) *"Q-learning"*. The firm applies the Q-learning framework to solve the Markov Decision Problem. The details of the Q-learning algorithm are described in the supplement, available online.

Evaluating Dynamic Reward Algorithms. We evaluate these algorithms using Yelp's social network [25] (Section 2.2) and a Facebook's social network with 4,038 users [17]. We consider

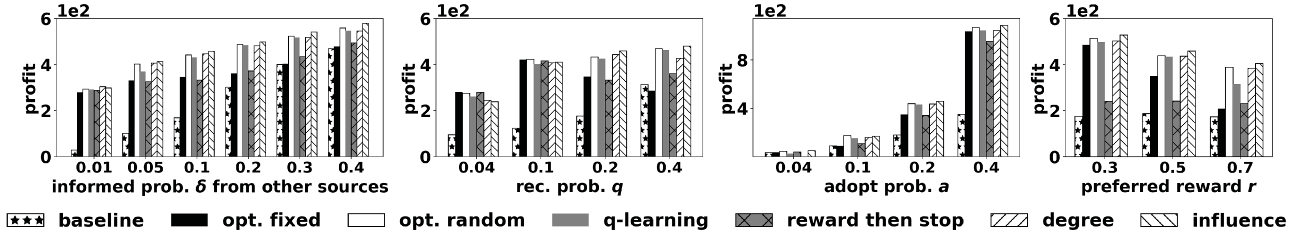


Fig. 23. Firm's total Profits for different algorithms (Facebook graph, average of 200 runs).

that the firm has a preferred amount of rewards (e.g., 22 GB storage space for Dropbox). The detailed settings and the algorithms are in the supplement file, available online.

• *Dynamic rewarding with model parameters.* We run each algorithm in Table 9 for 200 times to estimate a firm's expected profit. Fig. 23 shows the profit on Facebook's graph. First, we observe that randomizing the reward and price can further improve the firm's profit over the fixed action, especially when the preferred reward r_i is large. This is because this randomization can reduce a firm's expenses on rewards by giving rewards to only a fraction of users. Second, the state-dependent strategies "reward and then stop" or "Q-learning" cannot further improve a firm's profits compared to "random action". The indicates that the optimal strategies do not change a lot when users' states change. Third, the user-dependent strategies such as "high degree" or "high influence" can further improve a firm's profit compared to "random action". We have similar observations on Yelp's graph as shown in Fig. 24.

• *Choosing the "k" for influence maximization.* Influence maximization (IM) problem [14] is well studied that aims to select k "seed users" to give rewards, to maximize the influence of a product. However, the IM problem does not consider how to choose the proper value of k. Our *"high influence"* algorithm applies the IM algorithms to select k users to give rewards, and tune k to optimize the firm's profit. Fig. 20 shows that by tuning k, a firm can improves its profit by 5%$\sim$71% compared with the strategy to uniformly randomly choose k.

8 Related Works

The spread of word-of-mouth in online social network has been studied extensively. Domingos and Richardson [11], [21] studied the marketing problem of rewarding the customers in the social network. Many works studied influence maximization problem [7], [14], [23] that aims to select a set of seed users (to give rewards), to diffuse the product's information via word-of-mouth. However, these works did not address the problem of how to choose the number of seed nodes k because they did not model the firm's profit. Our model can be used to decide the optimal k, as well as the optimal rewards to offer to the recommenders. Our experiments (Fig. 20) show that the choice of the number of seed nodes k has a great impact on firm's profit. Various works proposed models to predict and forecast information cascades in networks [13], [16], [26]. Our model extends the Independent Cascade (IC) model [13], [14] for information's diffusion in OSNs. In particular, our model considers users' decisions on whether to adopt or recommend an item, which significantly reduces the error to predict the number of recommenders compared to the IC model. We like to point out that usually we do not have the metadata about the economic incentives for social recommendations such as the price and reward. In such cases, one should consider the IC model and others that do not require these metadata [16].

Social recommendation scheme was studied from an economic perspective (a.k.a. referral reward programs). Authors in paper [15] studied firms' optimal strategies. Campbell [5] studied impacts of word-of-mouth communication on firms' profits. Our work differs from these works, because our model includes important factors observed from data and we validate our model using data.

There is a rich literature about online decision making and reinforcement learning. Auer *et al.* in 2002[1] did finite-time analysis on the upper confidence bound (UCB) algorithm for the MAB problem. Chu *et al.* [9] consider the contextual MAB problem where the decision maker have extra contextual information. Posterior sampling (a.k.a. Thompson sampling) has provable performance for general reinforcement learning problem [19]. Chen *et al.* [8] studied the combinatorial MAB problem that can deal with the online influence maximization problem. To the best of our knowledge, we are the first to apply the posterior sampling reinforcement learning framework to decide firms' rewards and prices for social recommendation.

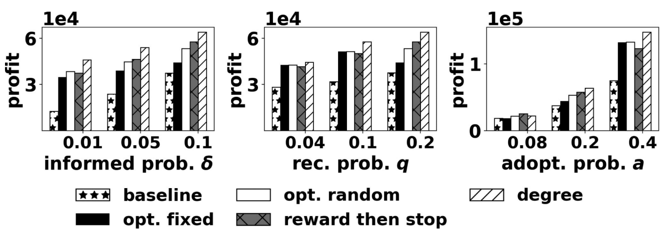


Fig. 24. Firm's total profits (Yelp, average of 200 runs).

9 Conclusion

This paper develops a data analytical framework to study social recommendations. We build mathematical models that are inspired by observations from data, and are calibrated with parameters inferred from data. Our model shows that rewarding recommenders can improve a firm's profit when the item is not well-known, or users' recommending probability is near a critical value for outbreak of diffusion. We also uncover conditions under which social recommendation may improve or hurt users' utilities. Moreover, we desing dynamic rewarding algorithms for a firm to dynamically improve its rewarding strategies.

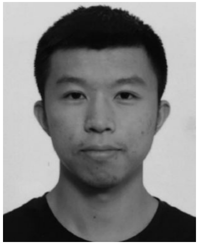
A firm can use our model to check whether the social recommendation scheme is profitable, and use our online algorithms to optimize its strategies. The OSN platform can use our model to evaluate the impact of social recommendations on its eco-system, and design regulating policies so that both firms and users can have benefits.

ACKOWNLEDGMENTS

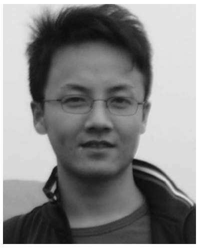
The work of John C.S. Lui was supported in part by the GRF 14200117. The work of Hong Xie was supported by Chongqing Natural Science Foundation (cstc2020jcyj-msxmX0652), Chongqing High-Technology Innovation and Application Development Funds (cstc2019jscx-msxmX0422, cstc2019jscx-fxyd0385), and the Fundamental Research Funds for the Central Universities (2020CDJ-LHZZ-057).

REFERENCES

[1] P. Auer, N. Cesa-Bianchi, and P. Fischer, “Finite-time analysis of the multiarmed bandit problem,” *Mach. Learn.*, vol. 47, no. 2/3, pp. 235–256, 2002.
[2] BBC, “Wechat hits one billion monthly users - are you one of them?” 2018. [Online]. Available: https://www.bbc.com/news/business-43283690
[3] E. Biyalogorsky, E. Gerstner, and B. Libai, “Customer referral management: Optimal reward programs,” *Marketing Sci.*, vol. 20, no. 1, pp. 82–95, 2001.
[4] D. S. Callaway, M. E. Newman, S. H. Strogatz, and D. J. Watts, “Network robustness and fragility: Percolation on random graphs,” *Phys. Rev. Lett.*, vol. 85, no. 25, 2000, Art. no. 5468.
[5] A. Campbell, “Word-of-mouth communication and percolation in social networks,” *Amer. Econ. Rev.*, vol. 103, no. 6, pp. 2466–2498, 2013.
[6] T. Chen and C. Guestrin, “XGBoost: A scalable tree boosting system,” in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2016, pp. 785–794.
[7] W. Chen, C. Wang, and Y. Wang, “Scalable influence maximization for prevalent viral marketing in large-scale social networks,” in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2010, pp. 1029–1038.
[8] W. Chen, Y. Wang, and Y. Yuan, “Combinatorial multi-armed bandit: General framework and applications,” in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 151–159.
[9] W. Chu, L. Li, L. Reyzin, and R. Schapire, “Contextual bandits with linear payoff functions,” in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 208–214.
[10] P. Davidsson, “Agent based social simulation: A computer science view,” *J. Artif. Societies Soc. Simul.*, vol. 5, no. 1, pp. 1–7, 2002.
[11] P. Domingos and M. Richardson, “Mining the network value of customers,” in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2001, pp. 57–66.
[12] Dropbox, How to earn more space by inviting friends, 2018. [Online]. Available: https://www.dropbox.com/help/space/earn-space-referring-friends
[13] A. Goyal, F. Bonchi, and L. V. Lakshmanan, “Learning influence probabilities in social networks,” in *Proc. 3rd ACM Int. Conf. Web Search Data Mining*, 2010, pp. 241–250.
[14] D. Kempe, J. Kleinberg, and É. Tardos, “Maximizing the spread of influence through a social network,” in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2003, pp. 137–146.
[15] L. J. Kornish and Q. Li, “Optimal referral bonuses with asymmetric information: Firm-offered and interpersonal incentives,” *Marketing Sci.*, vol. 29, no. 1, pp. 108–121, 2010.
[16] J. Leskovec, L. A. Adamic, and B. A. Huberman, “The dynamics of viral marketing,” *ACM Trans. Web*, vol. 1, no. 1, 2007, Art. no. 5.
[17] J. Leskovec and J. J. Mcauley, “Learning to discover social circles in ego networks,” in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 539–547.
[18] M. Newman, *Networks*. London, U.K.: Oxford Univ. Press, 2018.
[19] I. Osband and B. Van Roy, “Why is posterior sampling better than optimism for reinforcement learning?,” in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 2701–2710.
[20] J. Pearl *et al.*, “Causal inference in statistics: An overview,” *Statist. Surv.*, vol. 3, pp. 96–146, 2009.
[21] M. Richardson and P. Domingos, “Mining knowledge-sharing sites for viral marketing,” in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2002, pp. 61–70.
[22] P. R. Rosenbaum and D. Rubin, “The central role of the propensity score in observational studies for causal effects,” *Biometrika*, vol. 70, no. 1, pp. 41–55, 1983.
[23] Y. Tang, Y. Shi, and X. Xiao, “Influence maximization in near-linear time: A martingale approach,” in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2015, pp. 1539–1554.
[24] Yelp, Check-in offer, 2018. [Online]. Available: biz.yelp.com/support/check_in_offers
[25] Yelp, Yelp dataset challenge, 2018. [Online]. Available: www.yelp.com/dataset/challenge
[26] L. Yu, P. Cui, F. Wang, C. Song, and S. Yang, “From micro to macro: Uncovering and predicting information cascading process with behavioral dynamics,” in *Proc. IEEE Int. Conf. Data Mining*, 2015, pp. 559–568.

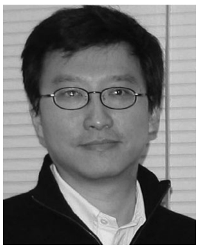


Li Ye (Member, IEEE) received the BEng degree from the School of Computer Science and Technology, University of Science and Technology of China, Hefei, China, in 2016. He is currently working toward the PhD degree in the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong, under the supervision of Prof. John C. S. Lui. His research interests include network economics, data driven decision, and stochastic modeling.



Hong Xie (Member, IEEE) received the BEng degree from the School of Computer Science and Technology, University of Science and Technology of China, Hefei, China, in 2010, and the PhD degree from the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong, in 2015, under the supervision of Prof. John C. S. Lui. He is currently a professor with the College of Computer Science, Chongqing University. His research interests include applied machine learning, data science, network economics, and stochastic modeling.

Yishi Lin received the BEng degree from the School of Computer Science and Technology, University of Science and Technology of China, Hefei, China, in 2013, and the PhD degree from the Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong, under the supervision of Prof. John C. S. Lui. She is currently a senior research with Tencent Inc. Her current research interests include social networks, such as information diffusion models and applications, and graph algorithms.



John C. S. Lui (Fellow, IEEE) received the PhD degree in computer science from the University of California at Los Angeles, Los Angeles, California. He was a chairman with the CSE Department from 2005 to 2011. He is currently the Choh-Ming Li chair professor with the Department of Computer Science and Engineering, Chinese University of Hong Kong. His current research interests include communication networks, system security (e.g., cloud security, mobile security, etc.), network economics, network sciences, large-scale distributed systems, and performance evaluation theory. He is an elected member of the IFIP WG 7.3, and a Croucher senior research fellow. He was a recipient of the various departmental teaching awards and the CUHK Vice-Chancellors Exemplary Teaching Award. He was also a co-recipient of the IFIP WG 7.3 Performance 2005 and IEEEIFIP NOMS 2006 best student paper awards. He serves in the editorial board of the *IEEE/ACM Transactions on Networking*, *IEEE Transactions on Computers*, *IEEE Transactions on Parallel and Distributed Systems*, *Journal of Performance Evaluation*, and the *International Journal of Network Security*.