Topic-aware Incentive Mechanism for Task Diffusion in Mobile Crowdsourcing through Social Network

JIA XU, YUANHANG ZHOU, GONGYU CHEN, and YUQING DING, Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, Nanjing University of Posts and Telecommunications, China
DEJUN YANG, Colorado School of Mines, USA
LINFENG LIU*, Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, Nanjing University of Posts and Telecommunications, China

Crowdsourcing has become an efficient paradigm to utilize human intelligence to perform tasks which are challenging for machines. Many incentive mechanisms for crowdsourcing systems have been proposed. However, most of existing incentive mechanisms assume that there are sufficient participants to perform crowdsourcing tasks. In large-scale crowdsourcing scenarios, this assumption may be not applicable. To address this issue, we diffuse the crowdsourcing tasks in social network to increase the number of participants. To make the task diffusion more applicable to crowdsourcing system, we enhance the classic *Independent Cascade* model so that the influence is strongly connected with both the types and topics of tasks. Based on the tailored task diffusion model, we formulate the *Budget Feasible Task Diffusion* (*BFTD*) problem for maximizing the value function of platform with constrained budget. We design a parameter estimation algorithm based on *Expectation Maximization* algorithm to estimate the parameters in proposed task diffusion model. Benefitting from the submodular property of the objective function, we apply the budget feasible incentive mechanism, which satisfies desirable properties of computational efficiency, individual rationality, budget feasible, truthfulness and guaranteed approximation, to stimulate the task diffusers. The simulation results based on two real-world datasets show that our incentive mechanism can improve the number of active users and the task completion rate by 9.8% and 11% averagely.

CCS Concepts: • **Networks** → Network algorithms; Network economics; • **Theory of computation** → Theory and algorithms for application domains; Algorithmic game theory and mechanism design.

Additional Key Words and Phrases: mobile crowdsourcing, social network, incentive mechanism, reverse auction, EM algorithm

ACM Reference Format:
Jia Xu, Yuanhang Zhou, Gongyu Chen, Yuqing Ding, Dejun Yang, and Linfeng Liu. 2021. Topic-aware Incentive Mechanism for Task Diffusion in Mobile Crowdsourcing through Social Network. *ACM Trans. Internet Technol.* 37, 4, Article 111 (August 2021), 23 pages.

*Corresponding author

Authors' addresses: Jia Xu, xujia@njupt.edu.cn; Yuanhang Zhou, Q17010120@njupt.edu.cn; Gongyu Chen, 1218043216@njupt.edu.cn; Yuqing Ding, Q17010107@njupt.edu.cn, Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, Nanjing University of Posts and Telecommunications, 9 Wenyuan Rd, Nanjing, Jiangsu, China, 210023; Dejun Yang, djyang@mines.edu, Colorado School of Mines, 1500 Illinois St., Golden, CO, USA, 80401; Linfeng Liu, liulf@njupt.edu.cn, Jiangsu Key Laboratory of Big Data Security and Intelligent Processing, Nanjing University of Posts and Telecommunications, 9 Wenyuan Rd, Nanjing, Jiangsu, China, 210023.

1533-5399/2021/8-ART111 $15.00
https://doi.org/

1 INTRODUCTION

Nowadays, mobile devices like smart phones have become popular and common in everyday life. Through the embedded sensors of mobile devices, people are with the ability to sense data of the surrounding environment, such as air pollution, noise level and share them through social networks. Mobile crowdsourcing has the advantages of low cost, low knowledge requirements, high flexibility, etc., and has been widely used in many fields such as environment (e.g. Safecast [40]), translation (e.g. Proz [37]), disaster response [44], and online marketplace (e.g. Freelancer [15]) in recent years.

Incentive mechanism is essential to mobile crowdsourcing since the smartphone users spend their time and consume battery, memory, computing power and data usage of device to generate, store and transmit the sensing data. Moreover, there are potential privacy threats to smartphone users by sharing their data with location tags, interests or identities. The issues of privacy leakage for mobile crowdsourcing systems [48] promote the research on incentive mechanism design for mobile crowdsourcing systems. Many existing research designs incentive mechanisms to attract the participants, stimulating users through monetary payment [54, 55, 57]. Besides, incentive mechanisms help the platform to select high-quality users, which improves the quality of crowdsourcing service [39, 56].

However, a crucial issue of mobile crowdsourcing is insufficient participants. Our statistics data showed that there are 21.1 uncompleted requests that were publicized more than 2 weeks in Amazon Mechanical Turk [1] on average from 2021-5-19 to 2021-5-30, while each request may include several HITs (Human Intelligence Tasks). Among these requests, 79.3% requests were publicized more than one month. Another observation from Freelancer [15] from 2021-5-19 to 2021-5-30 showed that there are 51.9 uncompleted projects that were publicized more than 2 weeks on average. Among them, 60.1% projects were publicized more than one month. The above surveys reveal the insufficient participant problem of current crowdsourcing systems.

To address this issue, we diffuse crowdsourcing tasks in the social network through registered users of crowdsourcing platform so that more users can be aware of and participate in the crowdsourcing. Different from many existing incentive mechanisms, which select winners to perform tasks, our incentive mechanism aims at selecting diffusers from the registered users to diffuse tasks to others in social network.

For the implementation of task diffusion for crowdsourcing, the diffusion model is needed to describe how the selected diffusers propagate their influence to other social users. Kempe *et al.* [22] proposed the two most popular influence diffusion models: *Independent Cascade* (*IC*) model and *Linear Threshold* (*LT*) model. A lot of studies have been made for influence maximization based on different diffusion models [10, 11], which provide the base of our work.

Further, we need to evaluate the influence of registered users. Most of existing influence calculation methods, such as degree centrality [6], k-shell decomposition [24], betweenness centrality [4], closeness centrality [26], evaluate influence of nodes from the point of network topology, and neglect the properties of specific items to be diffused. In mobile crowdsourcing, the registered users may diffuse multiple types of tasks simultaneously. The type of task represents what job needs to be done, e.g., translation, image recognition, speech acquisition, reading collection, fiction writing. The registered users probably have different influence on different types of tasks. For example, a novelist usually has higher influence for diffusing tasks of writing than diffusing tasks with other types. Thus, the classic methods of influence calculation are not applicable for crowdsourcing systems. Further, the influence of registered users is also related to the topics of tasks. The topics of tasks stand for the concrete content of tasks. For example, the topics of novel include horror fiction, love story, detective story, historical fiction, science fiction, etc. Usually, the users often have higher influence on topics in which they have reputation or are popular. For example, a science fiction

writer has higher influence on science fiction writing than writing on other topics. Therefore, the influence of a registered user should depend on not only the types of tasks it diffuses but also the topics of the tasks.

Taking all these issues into account, it is necessary to design a crowdsourcing task diffusion system to select the diffusers from the registered users of platform based on the influence on other social uses, and the appropriate incentive is desired to stimulate the task diffusers.

In this paper, we consider that the crowdsourcing tasks are launched by the platform, which is operated by some online community. So far, many online communities have developed crowdsourcing systems themselves, such as Stepes [42] operated by Facebook, Google Image Labeler [16] and Translate Community [43] operated by Google+, QQ-Crowd [38] operated by QQ, Crowdtesting [12] and Baidu Baike [2] operated by Baidu. We model the crowdsourcing task diffusion system as a reverse auction. In our system, each registered user of the platform decides on the task set it is willing to diffuse and submits its bid. Then the platform estimates the topics of crowdsourcing tasks and the influence of registered users. The platform selects winners and determines the payment. The winners diffuse the tasks to other users in the social network. Afterwards, the influenced social users perform the crowdsourcing tasks. Finally, each winner obtains the payment, which is determined by the platform. The objective of our incentive mechanism is designing truthful incentive mechanisms to maximize the value from the winners' task diffusion under the budget constraint. The whole process is illustrated by Fig. 1.

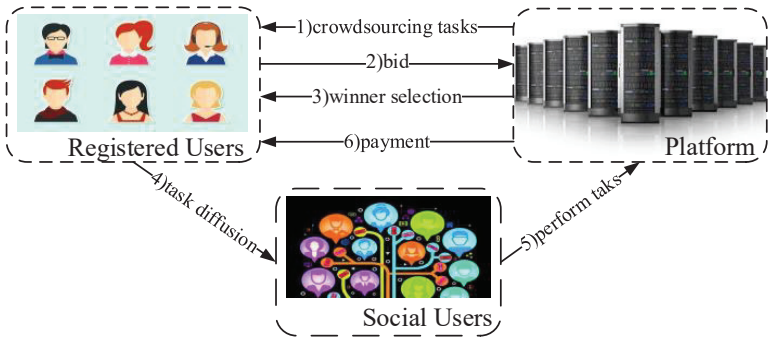

Fig. 1. Illustration of task diffusion in mobile crowdsourcing

The problem of designing truthful incentive mechanisms for crowdsourcing task diffusion is very challenging. First, the traditional influence diffusion model [22] should be enhanced so that the influence can be strongly connected with both the types and topics of tasks. Second, we need an efficient method to estimate the topics of multiple crowdsourcing tasks and the influence of registered users. Moreover, the registered user may take a strategic behavior by submitting dishonest bid price to maximize its utility.

The main contribution of this paper are as follows:

- To the best of our knowledge, this is the first work to design the truthful incentive mechanism for topic-aware task diffusion in crowdsourcing systems.
- We present the crowdsourcing task diffusion system and *Topic-aware Independent Cascade* (*TIC*) model, and formulate the *Budget Feasible Task Diffusion* (*BFTD*) problem to maximize the total value from task diffusion under the budget constraint.

- We present a *Parameter Estimation Algorithm* (*PEA*) to estimate the topics of crowdsourcing tasks and the influence of registered users for the task diffusion model based on the *Expectation Maximization* (*EM*) algorithm [13].
- We design an incentive mechanism, which satisfies desirable properties of computational efficiency, individual rationality, truthfulness and guaranteed approximation, to solve the *BFTD* problem. The simulation results based on two real-world datasets show that our incentive mechanism can improve the number of active users and the task completion rate by 9.8% and 11% averagely.

The rest of the paper is organized as follows. We review the state-of-art research in Section 2. Section 3 formulates the crowdsourcing model, diffusion model and problems, and lists some desirable properties. Section 4 presents the detailed design of parameter estimation algorithm for the diffusion model. Section 5 presents the detailed design of our incentive mechanism. Performance evaluation is presented in Section 6. We conclude this paper in Section 7.

2 RELATED WORK

2.1 Influence Diffusion and Influence Maximization

Diffusion in social network has been extensively studied in different fields as the basic for maximizing the influence. The pioneer work [22] proposed the classic *IC* diffusion model and *LT* diffusion model, and used a greedy algorithm to solve the problem of influence maximization. The concept of influence is specified as the expected number of active nodes at the end and has been used extensively. However, how to determine the influence probabilities in these two models is not mentioned.

Many improved diffusion models have been proposed thus far. Li *et al.* [28]studied diffusion dynamics and considered the different types of relationship between nodes in social network, including friendly and hostile ones, to maximize influence. Lu *et al.* [30] focused on the complementarity of different items and proposed the Comparative *Independent Cascade* model for influence maximization. [30] paid attention to the similarity between items, but did not give a specific classification. Doo *et al.* [14] designed a Probabilistic Social Influence model based on both *IC* model and *LT* model and studied how to use incentives to boost the diffusion. In [9], Chen *et al.* extended *IC* and *LT* models to incorporate time delay aspect.

Some work aims at estimating the influence of diffusion. The studies [4, 6, 24, 26, 35] estimate the influence only based on the metrics from the point of network topology, such as degree centrality [6], *k*-shell decomposition [24], betweenness centrality [4], closeness centrality [26], and generalized degree centrality [35]. These studies estimate the influence only based on the network topology. Therefore, the estimated influence cannot fit data well. Different from the above work, *EM* takes into account not only the network topology but also the historic data. Moreover, the time sequence of diffusion is also considered.

Saito *et al.* [41] first studied how to learn the influence probabilities in *IC* model, and applied *EM* algorithm to learn the parameters from history data. The strength of *EM* algorithm is that the results can fit data well. Goyal *et al.* [18] proposed the new models by considering the impact of time based on two classic models and defined the new metric termed influence score. In [18], the influence probabilities are calculated through history data and *Jaccard Index*. However, the method proposed in [18] requires the assumption of submodularity of joint influence probability, which is not always true. Kutzkov *et al.* [25] designed the randomized approximation algorithms to calculate influence based on data-stream and stable network topology in landmark and sliding window models. However, the influence calculated in [25] is self-defined and cannot be applied to our *TIC* model, which is tailored from the classic *IC* model.

Table 1. Comparison of Diffusion Influence Estimation Algorithms

Algorithms	[6]	[24]	[4]	[26]	[35]	[18]	[25]	*EM* [41]	*PEA*
using only topology	Yes	Yes	Yes	Yes	Yes	No	No	No	No
using topology	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
using historical data	No	No	No	No	No	Yes	Yes	Yes	Yes
using time sequence	No	No	No	No	No	Yes	Yes	Yes	Yes
diffusion model	None	None	None	None	None	*General Threshold Model*	None	*IC*	*IC*
hypothesis	None	None	None	None	None	submodularity	None	None	None
properties of items	No	No	No	No	No	No	No	No	Yes

Most importantly, all of the research mentioned above ignored the characteristics of items to be diffused, for example, the types and topics of tasks in our crowdsourcing context. In this paper, the classic *IC* model is tailored by extending the dimensionality of influence to task types and topics such that the diffusion model is more suitable for the crowdsourcing systems. We summarize the diffusion influence estimation algorithms in Table. 1.

2.2 Incentive Mechanisms for Social Crowdsourcing

Some recent research of crowdsourcing incentive mechanism design focused on the social network environment, in other words, the influence of participants towards the crowdsourcing system. Nie *et al.* [34] gave an overview on socially aware crowdsensing and designed an incentive mechanism based on *Bayesian Stackelberg* game. Zhao *et al.* [57] proposed a social-aware incentive mechanism based on social network effect and deep reinforcement learning for vehicular crowdsensing. Wang *et al.* [46] proposed a biased contest-based crowdsourcing system on social network to balance the sybil attack and heterogeneous effect of participants. Jiang *et al.* [21] designed incentive mechanisms based on time-sensitive and sybil-proof for a social network-based mobile crowdsensing system. Xiao *et al.* [50] focused on the task assignment problem in predictable mobile social networks and designed two algorithms to solve the problem. [8] investigated the structure of mobile sensing schemes and introduced crowdsourcing methods from the perspective of social network.

Some research focused on exploring the relationship between users in the social network. The influence of participants can be estimated based on their effects on social neighbors. Wang *et al.* [45] considered the worker recruitment in mobile crowdsensing system based on the influence propagation and designed two algorithms which have different efficiency to select winners. Xu *et al.* [52] proposed the online incentive mechanism for task diffusion through social network to solve the insufficient participation problem in crowdsourcing. Wang *et al.* [47] proposed a similar idea of task propagation via social network to stimulate task propagation and completion. Xu *et al.* [51] diffused the crowdsourcing tasks via the social network using the classic *IC* diffusion model and *LT* diffusion model and proposed influence estimation methods based on the topology and history knowledge. Xu *et al.* [53] designed truthful incentive mechanisms to minimize the social cost such that each of the cooperative tasks can be completed by a group of compatible users, where

the compatibility is modeled through the real-life relationships from social networks. Wang *et al.* [49] proposed a personalized task-oriented worker recruitment mechanism, where the tasks are allocated based on workers' preferences. Specifically, they modeled the initial preference of the new worker by averaging his social friends' preferences.

However, none of the above work design the topic-aware task diffusion model and corresponding the incentive mechanism for task diffusers in social network for crowdsourcing systems.

2.3 Topic Estimation for Crowdsourcing

Some studies have considered the topics of crowdsourcing tasks. Wang *et al.* [45] studied the problems of crowdsourcing worker recruitment and influence maximization in the social network, where the probability of worker accepting a crowdsourcing task depends on the topical interest and incentive attraction. Huang *et al.* [20] addressed the truth discovery problem considering topic relevance and truthfulness of claims as well as the topic awareness and reliability of sources. Ma *et al.* [31] estimated true value of observed variables in crowdsourced data by incorporating topic-specific expertise. However, the studies mentioned above don't estimate the topics of crowdsourcing tasks.

Topic estimation has been widely studied in other fields. Topic model is a type of statistical model for discovering the topics that occur in a collection of documents. An early topic model was described by Papadimitriou *et al.* in 1998 [36]. Another one, called *Probabilistic Latent Semantic Analysis* (*PLSA*), was created by Thomas in 1999 [19]. David *et al.* generalized *PLSA* and proposed *Latent Dirichlet allocation* (*LDA*) [3], which is the most common topic model currently in use. *LDA* introduces sparse Dirichlet prior distributions over document-topic and topic-word distributions, encoding the intuition that documents cover a small number of topics and that topics often use a small number of words. Other topic models are generally extensions on *LDA*, such as *Pachinko Allocation* [27], which improves *LDA* by modeling correlations between topics in addition to the word correlations which constitute topics. *Hierarchical Latent Tree Analysis* (*HLTA*) [29] is an alternative to *LDA*, which models word co-occurrence using a tree of latent variables and the states of the latent variables, which correspond to soft clusters of documents, are interpreted as topics.

However, the topics of crowdsourcing tasks usually cover vast domains of daily life. It requires a high semantic analysis ability in vast domains for the crowdsourcing platform to use the topic model. A low-cost method is employing *EM* algorithm to estimate the topics of tasks and the influence of registered users based on the historical data simultaneously.

3 SYSTEM MODEL

In this section, we model the crowdsourcing task diffusion system as a reverse auction. Then we present the *Topic-aware Independent Cascade model* and formulate the *Budget Feasible Task Diffusion* (*BFTD*) problem. Finally, we propose some desirable properties.

3.1 Crowdsourcing Task Diffusion System

We consider a crowdsourcing task diffusion system consisting of a platform and a set of registered users $U = \{1, 2, \ldots, n\}$, who are interested in diffusing crowdsourcing tasks. The platform publicizes a set $T = \{t_1, t_2, \cdots, t_m\}$ of m tasks, which need to be diffused in social network with the budget B. The budget B is the maximum value the platform plans to pay the selected registered users after they finish their diffusion work. The social graph $G = (V, E)$ includes all users in the social network. Each registered user $i \in U$ has its own social neighbors in social network. There can be at most m edges between any two nodes $v, w \in V$. For each edge $(v, w) \in E$, there is a weight indicating the influence of v on w for any task $t_j \in T$.

Usually, a crowdsourcing task is associated with multiple topics. For example, the mobile crowdsourcing task of collecting air pollution readings probably has the associated topics of climate,

environment and chemistry. These topics have different probabilities to reflect the content of crowdsourcing task (e.g., 50% for environment, 35% for climate and 15% for chemistry). Therefore, we use a probability distribution to represent the topics of a crowdsourcing task. We denote the topic of any task t_j as z_j, which follows a probability distribution over all topics, which is unknown to the platform, i.e., $\sum_{k \in Z} P\left(z_j = k\right) = 1$, where Z is the topic set.

Each registered user i submits a bid $\theta_i = (T_i, b_i)$, where $T_i \subseteq T$ is the task set registered user i willing to diffuse, and b_i is the bid price of i. Let c_i be the true cost of i, and c_i is private and only known to i.

Given the bid profile $\boldsymbol{\theta} = (\theta_1, \theta_2, \cdots, \theta_n)$, the platform selects a winner set $S \subseteq U$ and determines the payment δ_i for each registered user $i \in U$. The winners will be paid if they finish their diffusion work. Let $\boldsymbol{\delta} = (\delta_1, \delta_2, \cdots, \delta_n)$ be the payment profile.

We define the utility of registered user i as the difference between the payment and its real cost:

$$u_i = \begin{cases} \delta_i - c_i, & if\ i \in S \\ 0, & otherwise \end{cases} \tag{1}$$

Specifically, the utility of the losers would be zero because they are paid nothing and there is no cost for task diffusion.

Since we consider the registered users are selfish and rational individuals, each registered user can behave strategically by submitting a dishonest bid price to maximize its utility.

We denote $f(S)$ (will be characterized in section 3.2) as the value function of platform over the winner set S. The objective of our incentive mechanism is to maximize $f(S)$ under the budget B. We refer this problem as the *Budget Feasible Task Diffusion* (*BFTD*) problem, which can be formulated as follows:

$$Max\ f(S) \tag{2}$$

$$s.t.\ \sum\nolimits_{i \in S} \delta_i \leq B \tag{2-1}$$

3.2 Topic-aware Independent Cascade Model

Independent Cascade (*IC*) model is a classic model of influence diffusion, which has been widely studied. In *IC* model, a number of nodes become active at first time period. At any time period $\tau \in \mathbb{N}^+$, each active node v attempts to activate its neighbor node w with probability $p_{v,w} \in (0, 1)$. If the attempt succeeds, w becomes active at time period $\tau + 1$. If the attempt fails, v cannot activate w forever. We design our topic-aware task diffusion model based on *IC* model rather than the other classic LT model because it is hard to set the influence thresholds in LT model from the context of crowdsourcing. Furthermore, the *IC* model is closer to the realistic influence diffusion as the influence from one neighbor is independent of others' influence.

However, in crowdsourcing system, the influence between users is greatly connected with both the types and topics of tasks to be diffused. The classic *IC* model is not applicable. In our crowdsourcing task diffusion system, the active nodes at first time period are the winners of reverse auction, and each active user v attempts to activate its social neighbor w on task $t_j \in T_v$ with probability $p_{v,w}(j) \in (0, 1)$. The influence of nodes at time period $\tau > 0$ is the performance for diffusing tasks to social neighbors. The diffusion work can be viewed as the attempt to activate social users.

Furthermore, we present *Topic-aware Independent Cascade Model* (*TIC*) for our task diffusion. We consider that the registered users can always perform the tasks. Thus, we only diffuse the tasks to the unregistered users, i.e., users in V/U. The diffusion method of *TIC* is similar to that of *IC*, but the influence probability is related to the topics of tasks. For each pair $(v, w) \in E$, and topic $k \in Z$, there is a probability $p^k_{v,w}$, which indicates the influence of user v on its neighbor w about topic k.

Given a task t_j and its topic distribution $\gamma_j^k = P\{z_j = k\}$, $\sum_{k \in Z} \gamma_j^k = 1$, and $(v, w) \in E$, the probability of v to activate w successfully on task t_j can be calculated as:

$$p_{v,w}(j) = \sum_{k \in Z} \gamma_j^k p_{v,w}^k \tag{3}$$

For any user w, the activation probability of w on task t_j is:

$$p_w(j) = 1 - \prod\nolimits_{v \in N_w} \left(1 - p_{v,w}(j)\right) \tag{4}$$

where N_w is the neighbor set of w.

Given the winner set S, we denote the expected number of activated users on task t_j by $A_j(S)$. We can run Monte-Carlo simulations [33] of the *IC* model based on equations (3) and (4) for sufficiently many times (typically 1000) to obtain an accurate estimate of $A_j(S)$. Then we specialize the definition of $f(S)$ as the sum of expected number of activated users on all tasks through the task diffusion:

$$f(S) = \sum_{t_j \in T} A_j(S) \tag{5}$$

3.3 Desirable Properties

Our objective is to design an incentive mechanism satisfying the following desirable properties.

- **Computational efficiency:** An incentive mechanism $\mathcal{M}$ is computationally efficient if the winner set S and the payment profile $\boldsymbol{\delta}$ can be computed in polynomial time.
- **Individual Rationality:** Each registered user will have a non-negative utility when bidding its true cost, i.e. for $\forall i \in U$, $u_i \geq 0$.
- **Budget Feasibility:** An incentive mechanism is budget feasible if the total payment to the winners is not more than the budget, i.e. $\sum_{i \in S} p_i \leq B$.
- **Truthfulness:** An incentive mechanism is truthful if no registered user can improve its utility by submitting a false cost, no matter what others submit.
- **Approximation:** The objective of the mechanism is to maximize the value function of platform. We say that a mechanism is α-approximate if the mechanism outputs a winner set S such that $f(OPT) \leq \alpha f(S)$, where OPT is the optimal solution of *BFTD* problem.

The importance of the first three properties is obvious, because they together assure the feasibility of the incentive mechanism. The last two properties are indispensable for guaranteeing the compatibility and high performance. Being truthful, the incentive mechanisms can eliminate the fear of market manipulation and the overhead of strategizing over others for the registered users.

We list the frequently used notations in Table. 2.

4 TOPIC AND INFLUENCE ESTIMATION

In this section, we present a *Parameter Estimation Algorithm* (*PEA*) for our *TIC* model based on *EM* algorithm to estimate the topics of tasks and the influence of registered users.

In *TIC*, given the social graph $G = (V, E)$, topic set Z and history data $\mathbf{X}$, we need to estimate the parameters λ including the influence $p_{v,w}^k$ for each $(v, w) \in E$, $k \in Z$, and the topic distribution γ_j^k for each $t_j \in T$, $k \in Z$. We divide the history data $\mathbf{X}$ into m diffusion sets as $x_1, x_2, \cdots, x_m$, where $x_j = \left\{x_j(\tau) \middle| \tau \in \mathbb{N}^+\right\}$. $x_j(\tau)$ is the set of users who became active on task t_j at time period τ. Obviously, x_j shows the diffusion set of task t_j in the social graph.

EM algorithm is an iterative method to find the maximum likelihood estimation of parameters in statistical models, where the model depends on unobserved latent variables. In our model, we

Table 2. Frequently Used Notations

Symbol	Description
U, i, n	registered user set, registered user i, number of registered users
$G = (V, E)$	social graph
$T,\ T_i,\ t_j,\ m$	task set, task set of i, task t_j, number of tasks
$Z,\ k,\ z_j$	topic set, topic k, topic of task t_j
$\theta_i,\ \boldsymbol{\theta}$	bid of user i, bid profile
$b_i,\ c_i,\ u_i$	bid price of i, true cost of i, utility of i
B	budget
$\delta_i,\ \boldsymbol{\delta}$	payment to i, payment profile
$S,\ f(S)$	winner set, value function of platform
$\tau,\ \tau_j^w$	time period, time period that w became active on task t_j
$p_{v,w}(j)$	probability of v activating neighbor w on task t_j
$p_{v,w}^k$	influence of v on w about topic k
γ_j^k	topic distribution of task t_j
N_w	neighbor set of registered user w
$p_w(j)$	the activation probability of w on task t_j
$A_j(S)$	expected number of activated users on task t_j
$\lambda, \hat{\lambda}^a$	parameters in *TIC*, value of parameters λ after a iterations
$\mathbf{X}$	history data
$Q_j\left(k; \hat{\lambda}^a\right)$	posterior probability distribution of z_j
$l\left(\lambda; \hat{\lambda}^a\right)$	likelihood function
N_j^w	w's neighbors who had potential influence to w on task t_j
$\overline{N_j^w}$	w's neighbors who failed to influence w on task t_j
π_k	prior probability that any task's topic is k
$R_k^j(w, v; \lambda)$	probability that v activate the neighbor w successfully on task t_j with topic k
$f_i(S)$	marginal value of i over S
Φ	delay threshold
ε	convergence threshold

need to estimate parameters λ based on the history data $x_1, x_2, \cdots, x_m$, while the topics of tasks are hidden variables.

Given the history data $x_1, x_2, \cdots, x_m$, assuming that the diffusion set x_j is independent of the other diffusion sets, the likelihood of the history data, which is the joint probability of x_j, can be expressed as:

$$L(\lambda, \mathbf{X}) = L(x_1, x_2, \cdots, x_m; \lambda) = \prod_{j=1}^{m} P\left(x_j; \lambda\right) \tag{6}$$

where $P\left(x_j; \lambda\right)$ is the probability of the sample x_j when the parameters are λ.

The log-likelihood is:

$$l(\lambda, \mathbf{X}) = \log L(\lambda, \mathbf{X}) = \sum_{j=1}^{m} \log P\left(x_j; \lambda\right) \tag{7}$$

To find the maximum likelihood estimation of parameters λ, *EM* algorithm iteratively runs the following two steps until convergence.

E-step: Calculate the posterior probability distribution of z_j, which is topic of task t_j:

$$Q_j\left(k;\hat{\lambda}^a\right) = P\left(z_j = k|x_j,\hat{\lambda}^a\right),\ k \in Z \tag{8}$$

where $\hat{\lambda}^a$ is the value of parameters λ after a iterations.

Then calculate the likelihood function:

$$l\left(\lambda;\hat{\lambda}^a\right) = \sum_{j=1}^{m}\sum_{k\in Z} Q_j\left(k;\hat{\lambda}^a\right)\log P\left(x_j,k;\lambda\right) \tag{9}$$

M-step: Find the new estimation $\hat{\lambda}^{a+1}$ that maximizes the likelihood function:

$$\hat{\lambda}^{a+1} = \underset{\lambda}{\operatorname{argmax}}\, l\left(\lambda;\hat{\lambda}^a\right) \tag{10}$$

We extend the *EM* algorithm to estimate λ in *TIC* model by iterating the following two steps until convergence: a) Estimate the topic distribution of tasks via maximum likelihood estimation based on the newly calculated value of influence on each edge on the social graph; b) Estimate the value of influence on each edge based on new calculated topic distribution of tasks.

To find the likelihood of diffusion set x_j, we process the history data to differentiate the true diffusion path from the other potential paths. We denote the time period when w became active on task t_j as τ_j^w. If w was not active on t_j, we define $\tau_j^w = \infty$. To find out the path that w was influenced, we divide w's neighbors who were active on task t_j (active neighbors for short in the following) into two sets. We denote the set of w's neighbors who had potential influence to w on task t_j as N_j^w.

$$N_j^w = \begin{cases} \emptyset, & \tau_j^w = \infty \\ \left\{v|\,(v,w)\in E, 0 \le\ \tau_j^w - \tau_j^v \le \Phi\right\}, & otherwise \end{cases} \tag{11}$$

where Φ is a delay threshold.

We denote the set of w's neighbors who failed to influence w on task t_j as $\overline{N_j^w}$.

$$\overline{N_j^w} = \begin{cases} \emptyset, & \tau_j^w = \infty \\ \left\{v|\,(v,w)\in E,\ \tau_j^w - \tau_j^v > \Phi\right\}, & otherwise \end{cases} \tag{12}$$

Note that each active node has only one chance to active its neighbors in *TIC* model.

However, some active neighbors of w are not included in $N_j^w \cup \overline{N_j^w}$. These neighbors became active later than w, which means that they cannot influence w definitely. In other words, any neighbor $v \notin N_j^w \cup \overline{N_j^w}$ may be influenced by w, thus w must be included in the set of $N_j^v \cup \overline{N_j^v}$. Fig. 2. gives a toy example of neighbor division in a small social network with 7 nodes, where A and B became active at time period 1. C and D became active at time period 2. E became active at time period 3. F and G became active at time period 4. Thus, we have $x_j\,(1) = \{A,B\}$, $x_j\,(2) = \{C,D\}$, $x_j\,(3) = \{E\}$, $x_j\,(4) = \{F,G\}$. We set $\Phi = 1$ in this example. According to (11) and (12), we have $N_j^A = \{B\}$, $N_j^B = \{A\}$, $\overline{N_j^A} = \overline{N_j^B} = \emptyset$, $N_j^C = \{A,B,C\}$, $\overline{N_j^C} = \emptyset$, $N_j^D = \{B,C\}$, $\overline{N_j^D} = \emptyset$, $N_j^E = \{D\}$, $\overline{N_j^E} = \{A\}$, $N_j^F = \{E,G\}$, $\overline{N_j^F} = \{C\}$, $N_j^G = \{E,F\}$, $\overline{N_j^G} = \{D\}$.

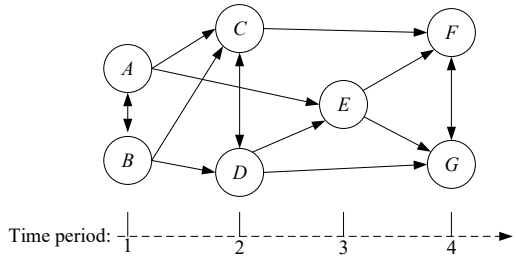

Fig. 2. An example of neighbor division in a small network, where the directed edges represent the task diffusion flows.

Intuitively, the likelihood of diffusion set x_j given topic k can be calculated as:

$$\begin{aligned}(x_j|k;\lambda) = &\prod_{w,\tau_j^w>1}\left(1-\prod_{v\in N_j^w}\left(1-p_{v,w}^k\right)\right)\left(\prod_{v\in\overline{N_j^w}}\left(1-p_{v,w}^k\right)\right)\\ &\prod_{w,\tau_j^w=0}\left(\prod_{v\in N_j^w}\left(1-p_{v,w}^k\right)\right)\left(\prod_{v\in\overline{N_j^w}}\left(1-p_{v,w}^k\right)\right)\end{aligned} \tag{13}$$

The joint distribution of diffusion set x_j and topic k can be calculated through conditional probability:

$$P\left(x_j,k;\lambda\right) = P\left(x_j|k;\lambda\right)P(k|\lambda) \tag{14}$$

We define $\pi_k = P(k|\lambda)$ as the prior probability that an arbitrary task's topic is k. The posterior probability distribution can be calculated as:

$$Q_j\left(k;\hat{\lambda}^a\right) = P\left(k|x_j,\hat{\lambda}^a\right) = \frac{P\left(x_j|k;\hat{\lambda}^a\right)\pi_k}{\sum_{\bar{k}\in Z}P\left(x_j|\bar{k};\hat{\lambda}^a\right)\pi_{\bar{k}}},\ k\in Z \tag{15}$$

Then we have the likelihood function:

$$l\left(\lambda;\hat{\lambda}^a\right) = \sum_{j=1}^{m}\sum_{k\in Z}Q_j\left(k;\hat{\lambda}^a\right)\log P\left(x_j,k;\lambda\right) \tag{16}$$

However, under this circumstance, it is impossible to maximize $l\left(\lambda;\hat{\lambda}^a\right)$ since $l\left(\lambda;\hat{\lambda}^a\right)$ is monotone over each $p_{v,w}^k$, and cannot be refined further. Fortunately, the likelihood of diffusion set x_j is an assumption we made before. We can redesign the likelihood function to make the maximization possible with a little loss of accuracy. We redesign the expression of likelihood of diffusion set x_j about topic k as:

$$P\left(x_j|k;\lambda\right) = \prod_{w}\left(\prod_{v\in N_j^w}p_{v,w}^{k\ R_k^j\left(v,w;\hat{\lambda}^a\right)}\left(1-p_{v,w}^k\right)^{1-R_k^j\left(v,w;\hat{\lambda}^a\right)}\right)\left(\prod_{v\in\overline{N_j^w}}\left(1-p_{v,w}^k\right)\right) \tag{17}$$

where

$$R_k^j(v,w;\lambda)=\begin{cases}0, & if\ v\notin N_j^w\\ \frac{p_{v,w}^k}{1-\prod_{\hat{v}\in N_j^w}\left(1-p_{\hat{v},w}^k\right)}, & otherwise\end{cases} \tag{18}$$

is the probability that v activate the neighbor w on task t_j and topic k successfully.

Then the likelihood function is:

$$\begin{aligned} l\left(\lambda;\hat{\lambda}^a\right) &= \sum_{j=1}^{m}\sum_{k\in Z} Q_j\left(k;\hat{\lambda}^a\right)\log P\left(x_j,k;\lambda\right)\\ &= \sum_{j=1}^{m}\sum_{k\in Z} Q_j\left(k;\hat{\lambda}^a\right)\log\pi_k\prod_w\left(\prod_{v\in N_j^w}{p_{v,w}^k}^{R_k^j\left(v,w;\hat{\lambda}^a\right)}\left(1-p_{v,w}^k\right)^{1-R_k^j\left(v,w;\hat{\lambda}^a\right)}\right)\left(\prod_{v\in\overline{N_j^w}}\left(1-p_{v,w}^k\right)\right) \end{aligned} \tag{19}$$

By this way, we can maximize the likelihood function easily via derivation, while the estimated parameters won't be affected significantly.

The method of learning parameters in *TIC* is shown in Algorithm 1, which follows the following four steps:

(1) Initialization: randomly generate π_k for all $k\in Z$ satisfying $\sum_{k\in Z}\pi_k=1$ and $p_{v,w}^k$ for all $(v,w)\in E$ (Line 1).

(2) Calculation of $Q_j\left(k;\hat{\lambda}^a\right)$ for $\forall k\in Z,\forall t_j\in T$and estimation of π_k for $\forall k\in Z$ (Lines 3-8).

(3) Calculation of $R_k^j\left(v,w;\hat{\lambda}^a\right)$ for $\forall k\in Z,\forall t_j\in T,\ \forall(v,w)\in E$ and estimation of $p_{v,w}^k$ for $\forall k\in Z,\forall(v,w)\in E$ (Lines 9-19).

(4) Convergence: The algorithm iterate step (2) and step (3) until the difference between the likelihood functions calculated using $\hat{\lambda}^{a+1}$ and $\hat{\lambda}^a$ is within a convergence threshold ε, i.e., $l(\lambda;\hat{\lambda}^{a+1})-l(\lambda;\hat{\lambda}^a)<\varepsilon$ (Line 2). Then we have the estimated $p_{v,w}^k$ and the value of γ_j^k given by $Q_j\left(k;\hat{\lambda}^a\right)$ (Lines 21-23).

5 INCENTIVE MECHANISM DESIGN

In this section, we present a *Budget Feasible Mechanism* (*BFM*) to solve the *BFTD* problem. We first explore some important properties of value function $f(S)$.

Definition 1. *(Nonnegative, monotone and submodular function): Given a finite ground set Ω, a real-valued set function defined as $2^\Omega\to\mathbb{R}$, F is called nonnegative, monotone and submodular if and only if it satisfies following conditions, respectively:*

$F(\emptyset)=0$ and $F(C)\geq 0$ for all C$\subseteq\Omega$;

$F(C)\leq F(D)$ for all $C\subseteq D\subseteq\Omega$;

$F(C\cup\{e\})-F(C)\geq F(D\cup\{e\})-F(D)$ for all $C\subseteq D\subseteq\Omega$ and any $e\in\Omega\backslash D$.

Theorem 1. *The value function $f(S)$ is nonnegative, monotone and submodular.*

Proof: Since $p_{v,w}(j)\in(0,1)$ for all $(v,w)\in E$, the nonnegativity of $f(S)$ is also obvious. The monotonicity of $f(S)$ is also obvious as adding a new user into S cannot decrease the value of $f(S)$.

Since a non-negative linear combination of submodular functions is also submodular, we only need to show $A_j(S)$ is submodular based on (5).

Consider a point of *IC* process when user attempts to activate its neighbor w on task t_j. The success probability $p_{v,w}(j)$ can be viewed as the random event determined by flipping a coin of

Algorithm 1 : Parameter Estimation Algorithm (PEA)

Require: social graph $G = (V, E)$, history data $\boldsymbol{X}$, task set T, topic set Z

Init $(\pi_k, p^k_{v,w})$;
while not convergence **do**
  //Estimation of π_k
  for all $k \in Z$ **do**
    for all $t_j \in T$ **do**
      $Q_j\left(k; \hat{\lambda}^a\right) \leftarrow \frac{P\left(x_j|k;\hat{\lambda}^a\right)\pi_k}{\sum_{\bar{k}\in Z} P\left(x_j|\bar{k};\hat{\lambda}^a\right)\pi_{\bar{k}}}$;
    end for
    $\pi_k \leftarrow \frac{1}{m} \sum_{t_j \in T} Q_j\left(k; \hat{\lambda}^a\right)$;
  end for
  //Estimation of $p^k_{v,w}$
  for all $k \in Z$ **do**
    for all $(v, w) \in E$ **do**
      for all $t_j \in T$ **do**
        $R^j_k\left(v, w; \hat{\lambda}^a\right) \leftarrow \frac{p^k_{v,w}}{1-\prod_{v\in N^w_j}\left(1-p^k_{v,w}\right)}$;
      end for
      if $v \notin N^w_j, \forall t_j \in T$ **then** $p^k_{v,w} = 0$;
      else
        $p^k_{v,w} \leftarrow \frac{\sum_{j,v\in N^w_j} Q_j\left(k;\hat{\lambda}^a\right)R^j_k\left(v,w;\hat{\lambda}^a\right)}{\sum_{j,v\in N^w_j\cup\overline{N^w_j}} Q_j\left(k;\hat{\lambda}^a\right)}$;
      end if
    end for
  end for
end while
for all $k \in Z$ **and** $t_j \in T$ **do**
  $\gamma^k_j \leftarrow Q_j\left(k; \hat{\lambda}^a\right)$;
end for
return $p^k_{v,w}, \gamma^k_j$, for $\forall v, w \in V, \forall k \in Z, \ \forall t_j \in T$

bias $p_{v,w}(j)$ independently. It is obvious that the outcome of *IC* process will not change if the coin is flipped at the beginning of the whole *IC* process.

Let Y_j denote the one sample point in the whole sample space in which each sample point is a possible set of outcomes for all coin flips of task t_j on all edges of social graph G. Let $A\left(S, Y_j\right)$ be the total number of users activated by the task diffusion when the winner set is S, and the set of outcomes of all coin flips on edges is Y_j. Let $H\left(v, Y_j\right)$ be the set of users that is reachable from v via at least one path consisting entirely of positive outcome edges under Y_j. We have:

$$A\left(S, Y_j\right) = \left|\cup_{v\in S} H\left(v, Y_j\right)\right| \tag{20}$$

Let any $C \subseteq D \subseteq U$ and $e \in U \backslash D$, we have:

$$\begin{aligned} &A\left(C \cup \{e\}, Y_j\right) - A\left(C, Y_j\right) \\ &= \left|H\left(e, Y_j\right) \setminus \left(\cup_{v\in C} H\left(v, Y_j\right)\right)\right| \end{aligned}$$

Algorithm 2 : Budget Feasible Mechanism (BFM)

Require: task set T, bid profile $\boldsymbol{\theta}$, budget B, the probability $p_{v,w}(j)$ for $\forall v \in U,\ \forall w \in V/U,\ \forall t_j \in T$

1: $S \leftarrow \emptyset$; $\boldsymbol{\delta} \leftarrow 0$; $S^* \leftarrow \{i | b_i \leq B\}$;
2: **with probability** 2/5:
3: $i^* \leftarrow \arg\max_{i \in S^*} f(\{i\})$;
4: $S \leftarrow \{i^*\}$; $\delta_{i^*} \leftarrow B$;
5: **with probability** 3/5:
//winner selection
6: $i \leftarrow \arg\max_{e \in S^*} \frac{f_e(S)}{b_e}$;
7: **while** $b_i \leq \frac{B \cdot f_i(S)}{2f(S \cup \{i\})}$ **do**
8: $S \leftarrow S \cup \{i\}$;
9: $i \leftarrow \arg\max_{e \in S^* \setminus S} \frac{f_e(S)}{b_e}$;
10: **end while**
//payment determination
11: **for all** $i \in S$ **do**
12: $S^{*\prime} \leftarrow S^* \setminus \{i\}$; $S' \leftarrow \emptyset$;
13: $i' \leftarrow \arg\max_{e \in S^{*\prime}} \frac{f_e(S')}{b_e}$;
14: **while** $b_{i'} \leq \frac{B \cdot f_{i'}(S')}{2 \cdot f(S' \cup \{i'\})}$ **do**
15: $i' \leftarrow \arg\max_{e \in S^{*\prime} \setminus S'} \frac{f_e(S')}{b_e}$;
16: $\delta_i \leftarrow \max\left\{\delta_i, \min\left\{\frac{B \cdot f_i(S')}{2f(S' \cup \{i\})}, \frac{f_i(S') \cdot b_{i'}}{f_{i'}(S')}\right\}\right\}$;
17: $S' \leftarrow S' \cup \{i'\}$;
18: **end while**
19: **end for**
20: **return** winner set S, payment profile $\boldsymbol{\delta}$

$$\geq \left|H\left(e, Y_j\right) \setminus \left(\cup_{v \in D} H\left(v, Y_j\right)\right)\right|$$

$$= A\left(D \cup \{e\}, Y_j\right) - A\left(D, Y_j\right)$$

Thus, we can conclude $A\left(S, Y_j\right)$ is submodular. Moreover, we have:

$$A_j(S) = \sum\nolimits_{Y_j} Prob\left(Y_j\right) A\left(S, Y_j\right) \tag{21}$$

Therefore, $A_j(S)$ is submodular since a non-negative linear combination of submodular functions is also submodular. ■

Considering the submodularity of $f(S)$, our *BFTD* problem falls into the study on *Budget Feasible Submodular Maximization Mechanism Design*. We design the *Budget Feasible Mechanism*, which is illustrated in Algorithm 2, based on Chen's random mechanism [7].

Let S^* denote the set of registered users with bid price no more than the budget B. With probability 2/5 (Lines 2-4), we select the registered user i^* with maximum value in set S^* as the winner, and the payment for i^* is equal to the budget.

With probability 3/5 (Lines 5-19), *BFM* performs the user selection step and payment determination step as follows:

In winner selection step (Lines 6-10), we process each registered user $i \in S^* \setminus S$ iteratively according to its marginal density $\frac{f_i(S)}{b_i}$, where $f_i(S)$ is the marginal value over set S of registered

user i, i.e. $f_i(S) = f(S \cup \{i\}) - f(S)$. In each iteration, if the bid price is not more than $\frac{B \cdot f_i(S)}{2f(S \cup \{i\})}$, we add the registered user i into the winner set.

In payment determination step (Lines 11-19), for each winner $i \in S$, we execute the winner selection step over $S^* \setminus \{i\}$ and denote the winner set as S'(Lines 14-18). We apply the *modified proportional share allocation rule* [52] to achieve the critical value of payment. The payment for any winner i is $\delta_i = max_{i' \in S'} \left\{ \min \left\{ \frac{B \cdot f_i(S'_{i'-1})}{2f(S'_{i'-1} \cup \{i\})}, \frac{f_i(S'_{i'-1}) \cdot b_{i'}}{f_{i'}(S'_{i'-1})} \right\} \right\}$, where $S'_{i'-1}$ is the winner set before we add i' into S'.

Lemma 1. *BFM is computationally efficient.*

Proof: The running time of *BFM* is dominated by the second branch (Lines 5-19). The maximum number of winners can be n. Calculating $f_e(S)$ takes $O(mn(|V| - n))$ time, thus finding the user with maximum marginal density (Line 6) takes $O\left(mn^2(|V| - n)\right)$ time. Hence, the while-loop (Lines 7-10) takes $O\left(mn^3(|V| - n)\right)$ time. In each iteration of the for-loop (Lines 11-19), a process similar to Lines 7-10 is executed. Hence, the payment determination takes $O\left(mn^4(|V| - n)\right)$. The running time of *BFM* is dominated by the payment determination step, which is bounded by $O\left(mn^4(|V| - n)\right)$. ∎

According to Lemma 1 and Corollary 3.5 in [23], we have the following theorem.

Theorem 2. *BFM is computationally efficient, individually rational, budget feasible, truthful, and has approximation ratio of 5.*

6 PERFORMANCE EVALUATION

We have conducted thorough simulations to investigate the performance of proposed algorithms, i.e., *PEA* (*Parameter Estimation Algorithm*) and *BFM* (*Budget Feasible Mechanism*). We use *PEA-BFM* to represent our incentive mechanism using *PEA* to estimate the parameters. We evaluate the performance of *PEA-BFM* against the following algorithms:

- *NPEA-BFM* (*Non-topic Parameter Estimation Algorithm [41] - Budget Feasible Mechanism*): This mechanism still adopts *EM* algorithm to estimate the parameters. However, it does not consider the topics of tasks.
- *TIE-BFM* (*Topology based Influence Estimation [51] - Budget Feasible Mechanism*): This mechanism uses the influence estimation method based on topology of the social network without considering the topics of tasks.
- *HIE-BFM* (*History based Influence Estimation [51] - Budget Feasible Mechanism*): This mechanism uses history data to estimate the influence of registered users without considering the topics of tasks.

We first measure the convergence of *PEA*. Then we measure the performance of four mechanisms with different number of registered users (n), number of tasks (m), and budget (B). All the simulations are run on a Centos 7 machine with Intel Xeon CPU E5-2420 and 16 GB memory. Each measurement is averaged over 100 instances.

6.1 Simulation Setup

We use two real-world datasets: Brightkite [5] and Gowalla [17] in our experiments with statistics given in Table 3. The datasets contain both relationship between network users and check-in data. The check-in records are in the form of <user id, check-in time, location>. We view each check-in as activeness on a task. For each dataset, we generate a random task id for each record.

The bid prices of registered users are randomly selected from the auction dataset [32], which contains 5017 bid prices for Palm Pilot M515 PDA from eBay. The bidding tasks of each registered user are randomly selected from the task set. The parameter settings are given in Table 4.

Table 3. Dataset Statistics

	Brightkite	**Gowalla**
Nodes	58228	196591
Edges	214078	950327
Check-ins	4491143	6442890

Table 4. Parameter Settings

Parameter	**Value**
$\|Z\|$	5
$\|V\|$	300
ε	0.001
n	100
m	80
B	15000
Φ	7 days
topic distribution	uniform distribution
number of tasks in each bid	[10, 20]

To measure the diffusion performance, we define two metrics, called number of active users and completion rate.

Number of active users: The algorithm outputs a winner set S. For each winner and each bidding task of this winner, we check her neighbors' check-in records on the task in the testing data. If a neighbor is not a registered user and has check-in records on the task, we count the neighbor as an active user. Each neighbor can be count only once. The total count of all active neighbors over all winners is the number of active users.

Completion rate: The algorithm outputs a winner set S. For each winner and each bidding task of this winner, we check her neighbors' check-in records on the task in the testing data. If a neighbor has check-in records on the task, we count the task once as a completed task. Each task can be count for multiple times. The completion rate is the rate of total count of completed tasks to the size of task set.

6.2 Convergence of *PEA*

We first measure the convergence of *PEA* through calculating the value of likelihood function $l(\lambda; \hat{\lambda}^a)$ at each iteration on both datasets. As shown in Fig. 3, the likelihood function increases rapidly at first, and then becomes stable after 50∼75 iterations in Brightkite, and 25∼50 iterations in Gowalla. In fact, we determine the value of convergence threshold ε based on these experiments. Since *PEA* can converge fast, we set a strict value of $\varepsilon = 0.001$, which is sufficient to guarantee the precision of *PEA*.

6.3 Impact of Number of Registered Users

To investigate the scalability of proposed mechanisms, we vary the number of registered users from 60 to 140. As shown in Fig. 4 and Fig. 5, the number of winners of all four algorithms increases with the increasing number of registered users. When more registered users can be selected, *BFM*

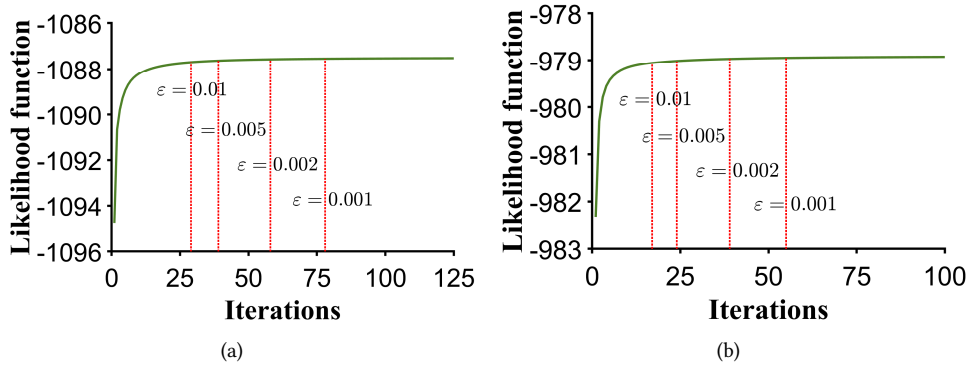

Fig. 3. Likelihood function versus iteration: (a) Brightkite. (b) Gowalla.

will select ones with larger marginal density, thus, under the fixed budget, *BFM* can select more users, and the value of platform of all four algorithms also increases.

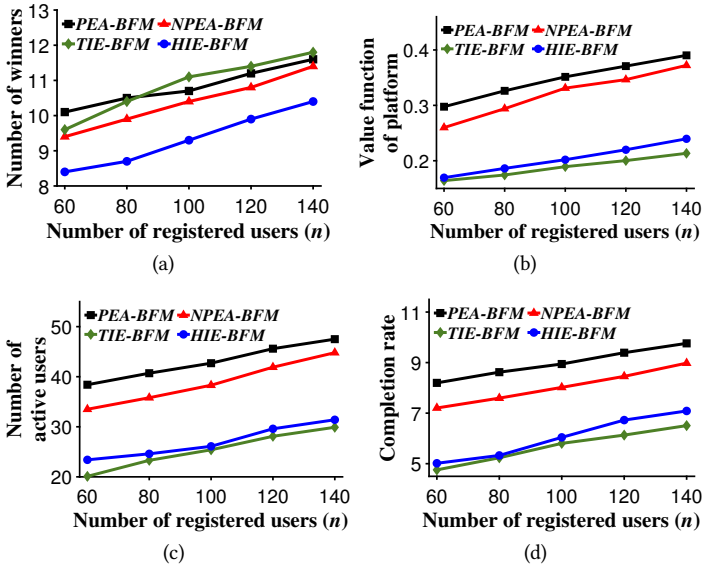

Fig. 4. Brightkite: Impact of number of registered users: (a) Number of winners. (b) Value function of platform. (c) Number of active users. (d) Completion rate.

The value of platform in *TIE-BFM* is much lower than that of *PEA-BFM* since *TIE-BFM* calculates the influence only based on the topology of the social network. The value of platform in *HIE-BFM* is also much lower than that of *PEA-BFM*. This is because the influence calculated by *HIE-BFM* only depends on the binary observation of history data. In *PEA-BFM*, we conduct the fine-grained process of history data by dividing the neighbors based on the delay threshold. Note that both *TIE-BFM* and *HIE-BFM* do not consider the topics of tasks. Thus, the influence calculated by them may be not accurate.

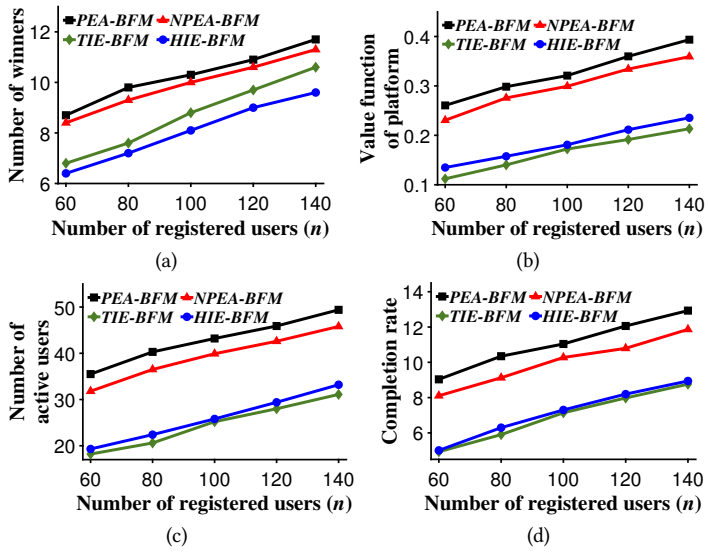

Fig. 5. Gowalla: Impact of number of registered users: (a) Number of winners. (b) Value function of platform. (c) Number of active users. (d) Completion rate.

The number of active users and completion rate reveal the effect of task diffusion. Similarly, the number of active users of four algorithms increase with increasing registered users. Among four algorithms, *PEA-BFM* has the best performance in terms of the number of active users and completion rate. *PEA-BFM* outperforms *NPEA-BFM*. This is because *PEA* considers the topics of tasks and influence probability over the topics, which makes the estimated parameters more accurate. Moreover, *HIE* outperforms *TIE* in all settings, but the difference is rather small. This is because *TIE* only consider the topology structure of social network, ignoring the history dataset.

6.4 Impact of Number of Tasks

Then, we vary the number of tasks from 40 to 120. Note that we measure the average the value function of platform, number of active users and completion rate over all winners. As shown in Fig. 6 and Fig. 7, *PEA-BFM* has the best performance among all algorithms. We find that the average contribution to the value of platform, number of active user and completion rate is decreasing with the increasing number of tasks. This is because when the number of tasks increases, the possibility that the bidding task sets of registered users have same tasks is less, reducing the influence on tasks. This leads to the loss of value of platform, number of active user and completion rate.

6.5 Impact of Budget

To investigate the impact of the budget, we vary the budget from 10000 to 20000. We can see from Fig. 8 and Fig. 9 that the number of winners increases with the increase of budget. Since the budget only affects the number of winners, the value of platform, number of active users and completion rate increase as well with increasing budget. *PEA-BFM* outperforms other three algorithms under all settings of budget.

Summary: *PEA-BFM* has the best performance for crowdsourcing task diffusion in all cases. *HIE-BFM* and *TIE-BFM* show low performance of task diffusion, comparing with the mechanisms based on *EM* algorithm. Comparing with the Non-topic mechanism, for Brightkite, *PEA-BFM* can

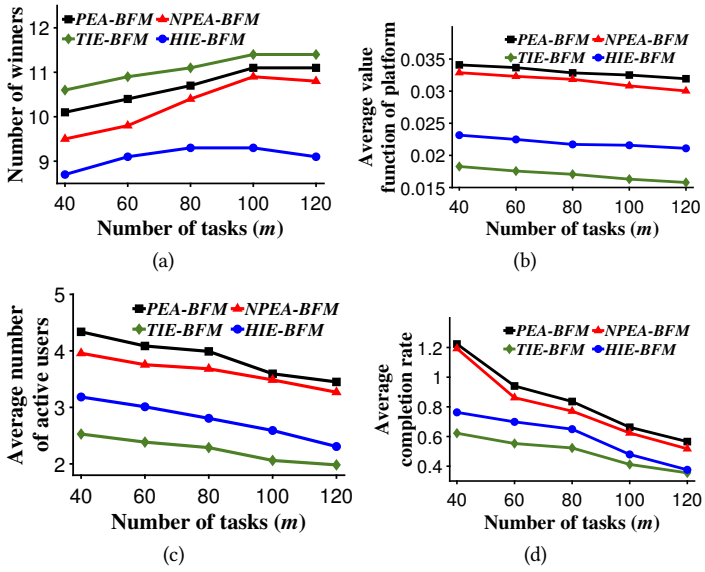

Fig. 6. Brightkite: Impact of number of tasks: (a) Number of winners. (b) Average value function of platform. (c) Average number of active users. (d) Average completion rate.

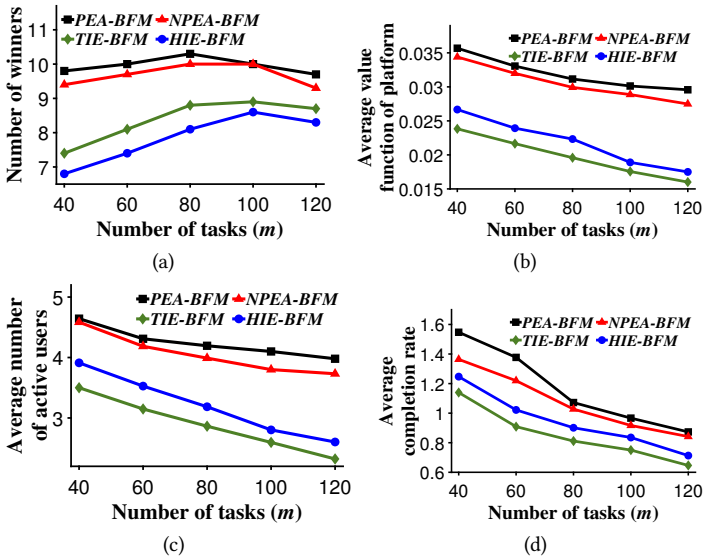

Fig. 7. Gowalla: Impact of number of tasks: (a) Number of winners. (b) Average value function of platform. (c) Average number of active users. (d) Average completion rate.

achieve average improvement of 8.28%, 10.60% and 11.59% in terms of value of platform, number of active users and completion rate, respectively. For Gowalla, the average improvement of *PEA-BFM* is 8.98%, 9.00% and 10.46%, respectively.

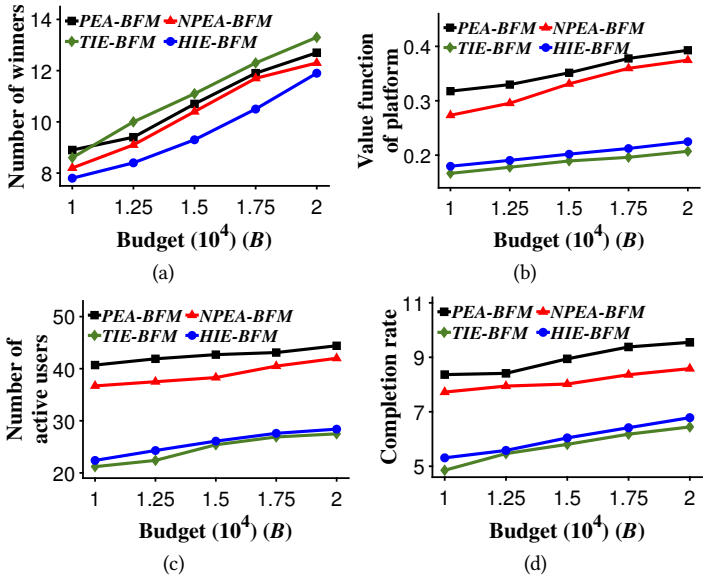

Fig. 8. Brightkite: Impact of budget: (a) Number of winners. (b) Value of platform. (c) Number of active users. (d) Completion rate.

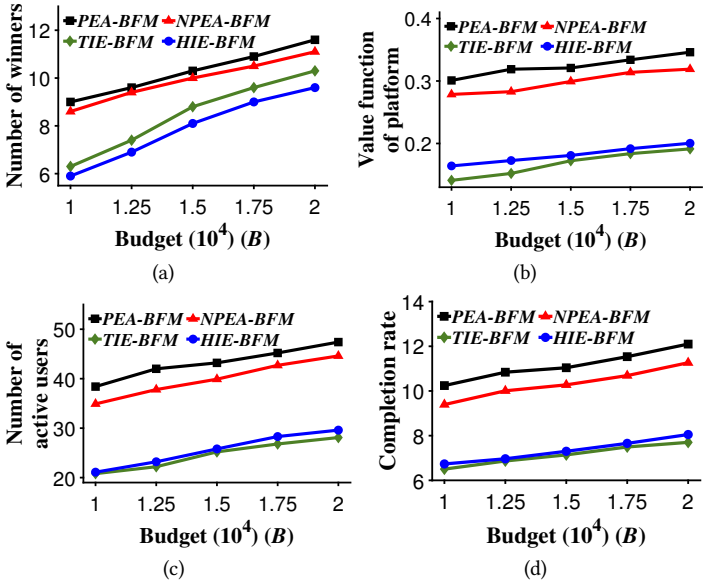

Fig. 9. Gowalla: Impact of budget: (a) Number of winners. (b) Value of platform. (c) Number of active users. (d) Completion rate.

7 CONCLUSION

In this paper, we have presented the mobile crowdsourcing task diffusion system and topic-aware independent cascade model. We have formulated the *BFTD* problem to maximize the total value from

task diffusion under the budget constraint. We have proposed a parameter estimation algorithm to estimate the topics of crowdsourcing tasks and the influence of registered users based on the *EM* algorithm. We have introduced the random budget feasible incentive mechanism, which satisfies desirable properties of computational efficiency, individual rationality, budget feasible, truthfulness and guaranteed approximation, to solve the *BFTD* problem. The simulation results based on two real-world datasets show that our incentive mechanism can largely increase the number of active users and improve the task completion rate.

ACKNOWLEDGMENTS

This work was supported in part by the NSFC grants 61872193, 61872191 and 62072254, and NSF grants 1717315.

REFERENCES

[1] AMT 2021. https://www.mturk.com
[2] Baidu Baike 2021. https://baike.baidu.com
[3] Blei, David, M., Ng, Andrew, Y., Jordan, Michael, I., and Lafferty. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3 (2003), 993–1022.
[4] Phillip Bonacich. 1972. Factoring and weighting approaches to status scores and clique identification. *Journal of mathematical sociology* 2, 1 (1972), 113–120.
[5] Brightkite 2021. http://snap.stanford.edu/data/loc-brightkite.html
[6] Duanbing Chen, Linyuan Lü, Ming-Sheng Shang, Yi-Cheng Zhang, and Tao Zhou. 2012. Identifying influential nodes in complex networks. *Physica a: Statistical mechanics and its applications* 391, 4 (2012), 1777–1787.
[7] Ning Chen, Nick Gravin, and Pinyan Lu. 2011. On the Approximability of Budget Feasible Mechanisms. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, Dana Randall (Ed.). SIAM, 685–699. https://doi.org/10.1137/1.9781611973082.54
[8] Pin-Yu Chen, Shin-Ming Cheng, Pai-Shun Ting, Chia-Wei Lien, and Fu-Jen Chu. 2015. When crowdsourcing meets mobile sensing: a social network perspective. *IEEE Commun. Mag.* 53, 10 (2015), 157–163. https://doi.org/10.1109/MCOM.2015.7295478
[9] Wei Chen, Wei Lu, and Ning Zhang. 2012. Time-Critical Influence Maximization in Social Networks with Time-Delayed Diffusion Process. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada*, Jörg Hoffmann and Bart Selman (Eds.). AAAI Press. http://www.aaai.org/ocs/index.php/AAAI/AAAI12/paper/view/5024
[10] Wei Chen, Chi Wang, and Yajun Wang. 2010. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, July 25-28, 2010*, Bharat Rao, Balaji Krishnapuram, Andrew Tomkins, and Qiang Yang (Eds.). ACM, 1029–1038. https://doi.org/10.1145/1835804.1835934
[11] Wei Chen, Yajun Wang, and Siyu Yang. 2009. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, June 28 - July 1, 2009*, John F. Elder IV, Françoise Fogelman-Soulié, Peter A. Flach, and Mohammed Javeed Zaki (Eds.). ACM, 199–208. https://doi.org/10.1145/1557019.1557047
[12] Crowdtesting 2021. http://test.baidu.com/crowdtest/community/index
[13] A. P. Dempster. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society* 39, 1 (1977), 1–22.
[14] M. Doo and L. Ling. 2014. Probabilistic Diffusion of Social Influence with Incentives. *IEEE Transactions on Services Computing* 7, 3 (2014), 387–400.
[15] Freelancer 2021. https://www.freelancer.com
[16] Google Image Labeler 2021. http://crowdsource.google.com/imagelabeler
[17] Gowalla 2021. http://snap.stanford.edu/data/loc-gowalla.html
[18] Amit Goyal, Francesco Bonchi, and Laks V. S. Lakshmanan. 2010. Learning influence probabilities in social networks. In *Proceedings of the Third International Conference on Web Search and Web Data Mining, WSDM 2010, New York, NY, USA, February 4-6, 2010*, Brian D. Davison, Torsten Suel, Nick Craswell, and Bing Liu (Eds.). ACM, 241–250. https://doi.org/10.1145/1718487.1718518
[19] Thomas Hofmann. 1999. Probabilistic Latent Semantic Indexing. In *SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 15-19, 1999, Berkeley, CA, USA*, Fredric C. Gey, Marti A. Hearst, and Richard M. Tong (Eds.). ACM, 50–57. https://doi.org/10.1145/312624.312649

[20] Chao Huang and Dong Wang. 2016. Topic-Aware Social Sensing with Arbitrary Source Dependency Graphs. In *15th ACM/IEEE International Conference on Information Processing in Sensor Networks, IPSN 2016, Vienna, Austria, April 11-14, 2016*. IEEE, 7:1–7:12. https://doi.org/10.1109/IPSN.2016.7460724

[21] Lingyun Jiang, Xiaofu Niu, Jia Xu, Yuchan Wang, Yongqi Wu, and Lijie Xu. 2018. Time-Sensitive and Sybil-Proof Incentive Mechanisms for Mobile Crowdsensing via Social Network. *IEEE Access* 6 (2018), 48156–48168. https://doi.org/10.1109/ACCESS.2018.2868180

[22] David Kempe, Jon M. Kleinberg, and Éva Tardos. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 24 - 27, 2003*, Lise Getoor, Ted E. Senator, Pedro M. Domingos, and Christos Faloutsos (Eds.). ACM, 137–146. https://doi.org/10.1145/956750.956769

[23] Pooya Jalaly Khalilabadi and Éva Tardos. 2018. Simple and Efficient Budget Feasible Mechanisms for Monotone Submodular Valuations. In *Web and Internet Economics - 14th International Conference, WINE 2018, Oxford, UK, December 15-17, 2018, Proceedings (Lecture Notes in Computer Science)*, George Christodoulou and Tobias Harks (Eds.), Vol. 11316. Springer, 246–263. https://doi.org/10.1007/978-3-030-04612-5_17

[24] Maksim Kitsak, Lazaros K. Gallos, Shlomo Havlin, Fredrik Liljeros, Lev Muchnik, H. Eugene Stanley, and Hernán A. Makse. 2010. Identification of influential spreaders in complex networks. *Nature Physics* (2010).

[25] Konstantin Kutzkov, Albert Bifet, Francesco Bonchi, and Aristides Gionis. 2013. STRIP: stream learning of influence probabilities. In *The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013, Chicago, IL, USA, August 11-14, 2013*, Inderjit S. Dhillon, Yehuda Koren, Rayid Ghani, Ted E. Senator, Paul Bradley, Rajesh Parekh, Jingrui He, Robert L. Grossman, and Ramasamy Uthurusamy (Eds.). ACM, 275–283. https://doi.org/10.1145/2487575.2487657

[26] Meizhu Li, Qi Zhang, and Yong Deng. 2018. Evidential identification of influential nodes in network of networks. *Chaos, Solitons & Fractals* 117 (2018), 283–296.

[27] Wei Li and Andrew McCallum. 2006. Pachinko allocation: DAG-structured mixture models of topic correlations. In *Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006 (ACM International Conference Proceeding Series)*, William W. Cohen and Andrew W. Moore (Eds.), Vol. 148. ACM, 577–584. https://doi.org/10.1145/1143844.1143917

[28] Yanhua Li, Wei Chen, Yajun Wang, and Zhi-Li Zhang. 2013. Influence diffusion dynamics and influence maximization in social networks with friend and foe relationships. In *Sixth ACM International Conference on Web Search and Data Mining, WSDM 2013, Rome, Italy, February 4-8, 2013*, Stefano Leonardi, Alessandro Panconesi, Paolo Ferragina, and Aristides Gionis (Eds.). ACM, 657–666. https://doi.org/10.1145/2433396.2433478

[29] Tengfei Liu, Nevin Lianwen Zhang, and Peixian Chen. 2014. Hierarchical Latent Tree Analysis for Topic Detection. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part II (Lecture Notes in Computer Science)*, Toon Calders, Floriana Esposito, Eyke Hüllermeier, and Rosa Meo (Eds.), Vol. 8725. Springer, 256–272. https://doi.org/10.1007/978-3-662-44851-9_17

[30] Wei Lu, Wei Chen, and Laks V. S. Lakshmanan. 2015. From Competition to Complementarity: Comparative Influence Diffusion and Maximization. *Proc. VLDB Endow.* 9, 2 (2015), 60–71. https://doi.org/10.14778/2850578.2850581

[31] Fenglong Ma, Yaliang Li, Qi Li, Minghui Qiu, Jing Gao, Shi Zhi, Lu Su, Bo Zhao, Heng Ji, and Jiawei Han. 2015. FaitCrowd: Fine Grained Truth Discovery for Crowdsourced Data Aggregation. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*, Longbing Cao, Chengqi Zhang, Thorsten Joachims, Geoffrey I. Webb, Dragos D. Margineantu, and Graham Williams (Eds.). ACM, 745–754. https://doi.org/10.1145/2783258.2783314

[32] Modeling online auctions 2021. http://www.modelingonlineauctions.com/datasets

[33] Christopher Z. Mooney. 1997. *Monte Carlo Simulation.* Sage Publications, Newbury Park, CA.

[34] Jiangtian Nie, Jun Luo, Zehui Xiong, Dusit Niyato, Ping Wang, and Mohsen Guizani. 2019. An Incentive Mechanism Design for Socially Aware Crowdsensing Services with Incomplete Information. *IEEE Commun. Mag.* 57, 4 (2019), 74–80. https://doi.org/10.1109/MCOM.2019.1800580

[35] Tore Opsahl, Filip Agneessens, and John Skvoretz. 2010. Node centrality in weighted networks: Generalizing degree and shortest paths. *Soc. Networks* 32, 3 (2010), 245–251. https://doi.org/10.1016/j.socnet.2010.03.006

[36] Christos H. Papadimitriou, Prabhakar Raghavan, Hisao Tamaki, and Santosh S. Vempala. 1998. Latent Semantic Indexing: A Probabilistic Analysis. In *Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 1-3, 1998, Seattle, Washington, USA*, Alberto O. Mendelzon and Jan Paredaens (Eds.). ACM Press, 159–168. https://doi.org/10.1145/275487.275505

[37] Proz 2021. https://www.proz.com

[38] QQ-Crowd 2021. http://task.qq.com

[39] Francesco Restuccia, Nirnay Ghosh, Shameek Bhattacharjee, Sajal K. Das, and Tommaso Melodia. 2017. Quality of Information in Mobile Crowdsensing: Survey and Research Challenges. *ACM Trans. Sens. Networks* 13, 4 (2017),

34:1–34:43. https://doi.org/10.1145/3139256

[40] Safecast 2021. https://safecast.org

[41] Kazumi Saito, Ryohei Nakano, and Masahiro Kimura. 2008. Prediction of Information Diffusion Probabilities for Independent Cascade Model. In *Knowledge-Based Intelligent Information and Engineering Systems, 12th International Conference, KES 2008, Zagreb, Croatia, September 3-5, 2008, Proceedings, Part III (Lecture Notes in Computer Science)*, Ignac Lovrek, Robert J. Howlett, and Lakhmi C. Jain (Eds.), Vol. 5179. Springer, 67–75. https://doi.org/10.1007/978-3-540-85567-5_9

[42] Stepes 2021. https://www.stepes.com

[43] Translate Community 2021. https://translate.google.com/community

[44] Hao Wang, Chi Harold Liu, Zipeng Dai, Jian Tang, and Guoren Wang. 2021. Energy-Efficient 3D Vehicular Crowdsourcing for Disaster Response by Distributed Deep Reinforcement Learning. In *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, Feida Zhu, Beng Chin Ooi, and Chunyan Miao (Eds.). ACM, 3679–3687. https://doi.org/10.1145/3447548.3467070

[45] Jiangtao Wang, Feng Wang, Yasha Wang, Daqing Zhang, Leye Wang, and Zhaopeng Qiu. 2019. Social-Network-Assisted Worker Recruitment in Mobile Crowd Sensing. *IEEE Trans. Mob. Comput.* 18, 7 (2019), 1661–1673. https://doi.org/10.1109/TMC.2018.2865355

[46] Yufeng Wang, Wei Dai, Qun Jin, and Jianhua Ma. 2020. BciNet: A Biased Contest-Based Crowdsourcing Incentive Mechanism Through Exploiting Social Networks. *IEEE Trans. Syst. Man Cybern. Syst.* 50, 8 (2020), 2926–2937. https://doi.org/10.1109/TSMC.2018.2837165

[47] Zhibo Wang, Yuting Huang, Xinkai Wang, Ju Ren, Qian Wang, and Libing Wu. 2021. SocialRecruiter: Dynamic Incentive Mechanism for Mobile Crowdsourcing Worker Recruitment With Social Networks. *IEEE Trans. Mob. Comput.* 20, 5 (2021), 2055–2066. https://doi.org/10.1109/TMC.2020.2973958

[48] Zhibo Wang, Xiaoyi Pang, Jiahui Hu, Wenxin Liu, Qian Wang, Yanjun Li, and Honglong Chen. 2019. When Mobile Crowdsensing Meets Privacy. *IEEE Commun. Mag.* 57, 9 (2019), 72–78. https://doi.org/10.1109/MCOM.001.1800674

[49] Zhibo Wang, Jing Zhao, Jiahui Hu, Tianqing Zhu, Qian Wang, Ju Ren, and Chao Li. 2021. Towards Personalized Task-Oriented Worker Recruitment in Mobile Crowdsensing. *IEEE Trans. Mob. Comput.* 20, 5 (2021), 2080–2093. https://doi.org/10.1109/TMC.2020.2973990

[50] Mingjun Xiao, Jie Wu, Liusheng Huang, Ruhong Cheng, and Yunsheng Wang. 2017. Online Task Assignment for Crowdsensing in Predictable Mobile Social Networks. *IEEE Trans. Mob. Comput.* 16, 8 (2017), 2306–2320. https://doi.org/10.1109/TMC.2016.2616473

[51] Jia Xu, Gongyu Chen, Yuanhang Zhou, Zhengqiang Rao, Dejun Yang, and Cuihua Xie. 2021. Incentive Mechanisms for Large-Scale Crowdsourcing Task Diffusion Based on Social Influence. *IEEE Trans. Veh. Technol.* 70, 4 (2021), 3731–3745. https://doi.org/10.1109/TVT.2021.3063380

[52] Jia Xu, Chengcheng Guan, Haobo Wu, Dejun Yang, Lijie Xu, and Tao Li. 2018. Online incentive mechanism for mobile crowdsourcing based on two-tiered social crowdsourcing architecture. In *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 1–9.

[53] Jia Xu, Zhengqiang Rao, Lijie Xu, Dejun Yang, and Tao Li. 2020. Incentive Mechanism for Multiple Cooperative Tasks with Compatible Users in Mobile Crowd Sensing via Online Communities. *IEEE Trans. Mob. Comput.* 19, 7 (2020), 1618–1633. https://doi.org/10.1109/TMC.2019.2911512

[54] Jia Xu, Jinxin Xiang, and Dejun Yang. 2015. Incentive Mechanisms for Time Window Dependent Tasks in Mobile Crowdsensing. *IEEE Trans. Wirel. Commun.* 14, 11 (2015), 6353–6364. https://doi.org/10.1109/TWC.2015.2452923

[55] Dejun Yang, Guoliang Xue, Xi Fang, and Jian Tang. 2012. Crowdsourcing to smartphones: incentive mechanism design for mobile phone sensing. In *The 18th Annual International Conference on Mobile Computing and Networking, Mobicom'12, Istanbul, Turkey, August 22-26, 2012*, Özgür B. Akan, Eylem Ekici, Lili Qiu, and Alex C. Snoeren (Eds.). ACM, 173–184. https://doi.org/10.1145/2348543.2348567

[56] Bowen Zhao, Shaohua Tang, Ximeng Liu, and Xinglin Zhang. 2021. PACE: Privacy-Preserving and Quality-Aware Incentive Mechanism for Mobile Crowdsensing. *IEEE Trans. Mob. Comput.* 20, 5 (2021), 1924–1939. https://doi.org/10.1109/TMC.2020.2973980

[57] Yinuo Zhao and Chi Harold Liu. 2021. Social-Aware Incentive Mechanism for Vehicular Crowdsensing by Deep Reinforcement Learning. *IEEE Trans. Intell. Transp. Syst.* 22, 4 (2021), 2314–2325. https://doi.org/10.1109/TITS.2020.3014263