IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received May 31, 2017, accepted June 29, 2017, date of publication July 25, 2017, date of current version August 22, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2731360

Fine-Grained Recommendation Mechanism to Curb Astroturfing in Crowdsourcing Systems

ZHIWEI GUO[1], CHAOWEI TANG[1], WENJIA NIU[2], YUNQING FU[3], TONG WU[4], HAIYANG XIA[5], AND HUI TANG[1]

[1]College of Communication Engineering, Chongqing University, Chongqing 400044, China
[2]Beijing Key Laboratory of Security and Privacy in Intelligent Transportation, Beijing Jiaotong University, Beijing 100044, China
[3]College of Software, Chongqing University, Chongqing 400044, China
[4]Tsinghua University, Beijing 100084, China
[5]School of Computer Science, Xi'an Shiyou University, Xi'an 710065, China

Corresponding author: Chaowei Tang (cwtang@cqu.edu.cn) and Wenjia Niu (nwj6688@gmail.com)

This work was supported by the National Natural Science Foundation of China under Grant 61672091.

ABSTRACT Crowdsourcing activities, carrying out large-scale tasks via wisdoms of crowds, are widely used in practice. However, it is hard for users to find tasks that are suitable for them. Thus, many users participate in tasks, and they are not good at or not interested in, and give answers carelessly or randomly. This phenomenon causes heavy astroturfing problem in crowdsourcing systems, which not only hurts the quality of completing tasks, but also influences user experience. Therefore, recommendation mechanisms that can optimize the match between users and tasks are in demand. However, existing studies simply adopt users' expertise level or interest degree as the key rule for recommendation. They neglect the fact that interest and expertise function jointly, and that interest can sometimes exert reaction force on expertise. Besides, previous studies assume that users' interest degree is steady, yet ignoring that it is time-varying rather than static in real world. In this paper, we propose IntexCrowd, fine-grained recommendation mechanism through interest-expertise collaborative awareness for crowdsourcing systems, to curb astroturfing problem. First, the IntexCrowd assigns a topic to each task. Then, topic-specific expertise level as well as interest degree of users are estimated according to historical records of tasks. At last, suitable user lists for topic-specific tasks are suggested as recommendation results. And we present a case study and a set of experiments to confirm the validity of IntexCrowd.

INDEX TERMS Crowdsoucing, astroturfing, ranking, interest degree, expertise level, collaborative awareness.

I. INTRODUCTION

With the development of online services, the application of using wisdoms of crowds to carry out large-scale tasks has been undergoing an unprecedented rise. Crowdsourcing [17] is an operation mode where requesters can call for workers with different capabilities to process tasks for monetary reward [2]. Its benefits have been widely recognized today [4], [27], [28] and more than one croudsourcing system such as Amazon Mechanical Turk[1] has been in operation. Major components of crowdsourcing systems in this paper contain service providers who operate the platforms, requesters who post tasks, and users who participate in tasks and give answers.

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

Despite much progress, some problems still remain challenging for service providers, in which astroturfing acts as a main one. "Astroturfing" initially refers to a type of online behaviors where astroturfers are hired to present certain beliefs or opinions that are mostly irresponsible. For example, when an e-commerce seller manages to promote sales of his own commodities, it is a practical way for him to hire astroturfers to fabricate comments beneficial to his commodities. The astroturfers just finish tasks to earn payoffs, regardless of objectivity of their remarks. In terms of crowdsourcing activities, astroturfers refer to users who complete tasks awfully or with high error rate. A main reason lies in that it is hard for users to find tasks that are suitable for them from massive ones. Thus, many users just participate in tasks they are not good at or not interested in, and give answers carelessly or randomly. Astroturfing problem not only hurts the quality of completing tasks, but also influences user experience.

To tackle the problem, efficient assignment or recommendation mechanisms became an urgent need [33]. In recent years, methods concerning selecting users for tasks or recommending tasks to users had been proposed. The most intuitive idea was to select users based on users' reliability [9], [10], [12], [15], [16], [24], [26], [29], [32], [40], which means that the higher frequency a user gives true answers, the more reliable he is. However, in these approaches, only one reliability was estimated for each user, which cannot properly reflect the variation in reliability among topics in real world. In fact, users' reliability usually varies among different topics. Having realized this, many researchers took division of topic domain into account. And two more representative measurements that are related to astroturfing problem were put into use: topical expertise [8], [18], [19], [22], [31], [34], [36], [38], [39], [41] and topical interest [3], [13], [20], [25], [30], [33], [35], [41]. Nevertheless, they are still not fine-grained enough.

On the one hand, almost all of these methods singly measured expertise level or interest degree as the principle for recommendation. They ignored the fact that interest degree and expertise level function jointly, and that interest degree may sometimes exert reaction force on expertise level. Specifically, interest degree may produce positive or negative feedback during the process of completing tasks, because relatively high or low interest degree will influence users' attitude towards working. We give an example shown in Fig. 1 to demonstrate this view. Fig. 1 demonstrates a virtual scene that is possible in real world. Let accuracy represent the quality of completing tasks by different users. User A and User C have close expertise level. If User A is not quite interested in these tasks as User C, he may not deal with tasks well as C does. Similarly, if User B has low expertise level but high interest degree, he may deal with tasks no worse than User A.

User	Expertise Level	Interest Degree	Accuracy
A	50	30	0.80
B	35	45	0.85
C	50	45	0.95

FIGURE 1. Interest can sometimes exert reaction force on expertise.

On the other hand, users' interest degree was assumed to be steady in all of these methods. But it is time-varying rather than static in real world, which is affected by moods, contexts, and pop culture trend.

In this paper, we focus on selecting appropriate users for multi-tag labelling tasks and propose Fine-grained Recommendation Mechanism through *Int*erest-*Ex*pertise Collaborative Awareness for *Crowd*sourcing Systems (**IntexCrowd**) to curb astroturfing problem. First, the proposed mechanism employs a topic model to assign each task a topic. Then, it considers the process of completing tasks as logistic regression of interest degree as well as expertise level, and adopts Bayesian posterior inference to estimate topic-specific interest degree as well as expertise level of each user. At last, a ranking algorithm through ''interest-expertise collaborative awareness`` is developed to suggest appropriate user lists for topic-specific tasks. To the best of our knowledge, we are the first to consider "interest-expertise collaborative awareness" when it comes to recommendation mechanism for curbing astroturfing problem in crowdsourcing systems. The main contributions of this paper are described as follows:

- In order to curb astroturfing problem in crowdsourcing systems more finely, the collaborative awareness of expertise level and interest degree is proposed as basis to recommend appropriate users for topic-specific tasks. Different from previous studies that singly consider expertise level or interest degree, we further consider the collaborative effect of the two factors.
- We assume that users' interest degree is time-varying rather than static, which is more closed to real-world scenarios.
- A specified case study and some experimental evaluations are conducted to demonstrate efficiency of the proposed IntexCrwod.

The rest of the paper is organized as follows. In Section II, we survey the related work. In Section III, we introduce IntexCrowd framework and functionality of its components. The fine-grained recommendation mechanism through interest-expertise collaborative awareness is proposed in Section IV. Section V presents the experimental results and analyses. Finally, Section VI concludes the paper.

II. RELATED WORK

In recent years, to curb astroturfing problem in crowdsourcing systems, recommendation mechanisms which can optimize the match between users and tasks become an important means. Accordingly, techniques concerning selecting users for tasks or recommending tasks to users have been proposed. Thus in this section, we introduce the prior works related to our research.

Some existing approaches select users with high reliability, which is the most intuitive idea. Cao *et al.* [10] proposed a crowd selection method by calculating error rate of users to obtain users' reliability. Raykar and Yu. [26] proposed an empirical Bayesian algorithm called SpEM that iteratively eliminates spammers and then estimates reliability of users from past behaviors. [12] presented a method Pick-a-Crowd that measures reliability through predicting accuracy of completing tasks. Ho *et al.* [15] and Ho and Vaughan [16] also estimated accuracy as reliability and assigned tasks to users on the basis of budget minimization under the premise of ensuring quality of completing tasks. Tran-Tranh *et al.* [29], modeled task assignment using bounded multi-armed bandits with the goal of maximizing the overall utility achieved. In [9], [24], [32], and [40], researchers proposed different methods to measure reliability

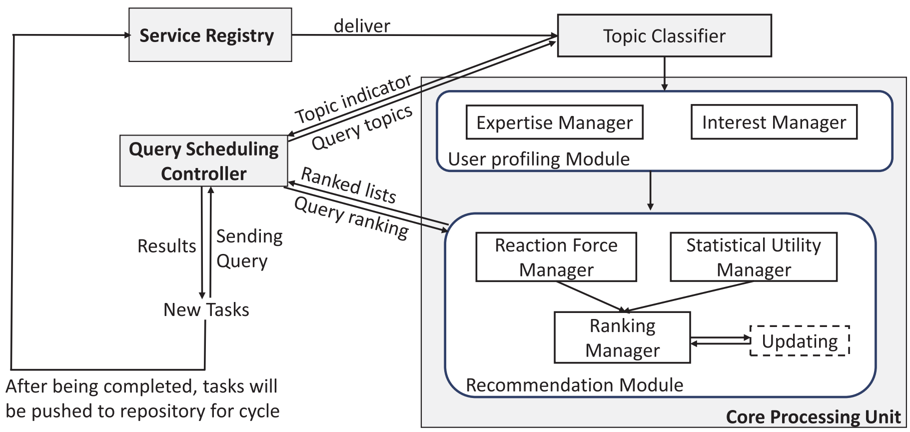


FIGURE 2. Infrastructure of IntexCrowd.

of users and to select appropriate users for tasks. Note that IntexCrwod is quite different from the above ones. Almost all of the above methods only focus on simple point such as reliability, yet neglect the fact that reliability is never constant and varies with different topics.

Having realized this, some researchers take division of topic domain into account. And two representative measurements that are related to astroturfing problem are put into use: topical expertise and topical interest. Bouguessa *et al.* [8], proposed a model to find experts based on Indegree that is the number of best answers provided by users. Jurczyk and Agichtein [18], applied HITS algorithm [19] on the underlying graph of question answering for estimating user ranking scores. Zhang *et al.* [34], proposed expertise ranking and evaluated link algorithms on a specific domain dataset. They also proposed Z-score to measure the relative expertise of a user. Yang *et al.* [31], presented the CQARank model to estimate both latent topics of questions and topical expertise by exploiting voting information. Zhao *etal.* et al. [36], proposed TEL model to generate experts and topics simultaneously by using historical contribution of users. Zhao *et al.* [38], developed a Bayesian generative model to exploit latent skills of users as well as the latent categories of tasks. Ma *et al.* [22], proposed FaitCrowd mechanism that jointly models the process of generating question content and providing answers in a probabilistic model to estimate topical expertise. Liu *et al.* [21] used mixture of language model and LDA to predict best answerers. In the meanwhile, some researchers started to exploit recommendation mechanism based on topical interest degree. Zhao *et al.* [39], tackled crowd selection problem by transferring the knowledge from categorized Yahoo! Answers datasets for learning users' interest degree. Guo *et al.* [13], proposed a generative model for question answering by exploring the category information to discover latent interests of users. Lin *et al.* [20], presented a method that enables a system to leverage implicit signals about task preferences and introduced classical collaborative filtering to recommend tasks to users. Xu *et al.* [30], proposed a DRM model that considers dual roles of users and then measured their interest degree for recommendation. And in [3], [25], [33], and [35], researchers almost proposed various methods to measure users' interest degree for recommendation. Although these approaches are more reasonable, they are still not fine-grained enough compared with IntexCrwod. First, these methods fail to consider the time-varying characteristics of interest degree. Second, almost all of these methods simply measure expertise level or interest degree as the key principle for recommendation and ignore the fact that interest degree and expertise level function jointly.

[31] and [41] are recent and relatively advanced researches, where researchers considered the joint effect of expertise level and interest degree. However, principles of their methods are simply weighted accumulation of expertise level and interest degree, thus still ignoring the fact that interest degree can sometimes exert reaction force on expertise level.

III. IntexCrowd FRAMEWORK

In this section, we describe our IntexCrowd framework shown in Fig. 2. The framework consists of four main components: service registry, query scheduling controller, topic classifier, and core processing unit. Among them, service registry is a repository that stores records of historical tasks records with users' participation. For tasks of service registry and newly released tasks, topic classifier will assign each of them a topic. The query scheduling controller can send a request to core processing unit to ask for recommendation results of newly released tasks according to their topic indicators. The core processing unit contains several managers and is divided into two modules, that is to say, a module is a manager group comprised by several managers. The core processing unit is responsible for reading historical records of specific topic, formulating user profile (expertise level and interest degree), and suggesting ranked lists of users for topic-specific tasks as recommendation results. Functionality of topic classifier and five managers in core processing unit is described as follows:

A. TOPIC CLASSIFIER

It is an interface provided to tasks, and functions as automatically assigning a topic to each task through topic model. It not only assigns each task in service registry a topic, but identifies topic of newly released tasks. As we assume that there are K topics, thus topic indicator k ranges from 1 to K.

B. EXPERTISE MANAGER AND INTEREST MANAGER

Expertise manager and interest manager work in parallel. The former is responsible for assessing users' topical expertise level, while the latter for assessing users' topical interest degree. Given topic k, all the historical records of topic k is delivered to the two managers to produce user profile. In real world, as users' interest degree usually drifts over time, it is essential to capture users' temporal interest degree values. We denote each time a user participates in a task as an interaction happening at a time point. Then, we assume that there is an instantaneous interest degree associated with occurrence of each interaction. Having divided topic domains, all instantaneous interest degree values of users can be clustered into K groups and sorted in chronological order respectively. Here we propose the following definitions:

Definition 1: expertise level, denoted as an array e_{ku}, refers to user u's level of reliability for K topics.

$$e_{ku} = [e_{1u}, e_{2u}, e_{3u}, \ldots, e_{Ku}]\,, \text{where } e_{ku} \in (-\infty, +\infty) \tag{1}$$

Definition 2: temporal interest degree, denoted as $\left\{i_{ku}^{(t)}\right\}$ who is an ensemble of K arrays, refers to user u's instantaneous interest degree at t different timestamps for K topics.

$$\underset{\text{(K topics)}}{i_{ku}^{(t)}} = \begin{cases} i_{1u}^{(t)} = \left[i_{1u}^{(1)}, i_{1u}^{(2)}, \ldots, i_{1u}^{(t)}\right] \\ i_{2u}^{(t)} = \left[i_{2u}^{(1)}, i_{2u}^{(2)}, \ldots, i_{2u}^{(t)}\right] \\ \cdots \\ i_{Ku}^{(t)} = \left[i_{Ku}^{(1)}, i_{Ku}^{(2)}, \ldots, i_{Ku}^{(t)}\right] \end{cases} \tag{2}$$

Definition 3: current interest degree, denoted as a vector I_{ku}, refers to user u's instantaneous interest degree for K topics at current timestamp, which is obtained through prediction from temporal interest degree.

$$I_{ku} = [I_{1u}, I_{2u}, \ldots, I_{Ku}] \tag{3}$$

C. REACTION FORCE MANAGER AND STATISTICAL UTILITY MANAGER

Recommendation module fetches estimated current interest degree and expertise level of users from profile module. Whereas, in real world, interest degree can sometimes produce positive feedback or negative feedback during the process of completing tasks, because relatively high or low interest degree will influence users' attitude towards tasks. Besides, as topical interest degrees are usually drifting in the real world, we hold the opinion that interest degree with more positive statistical trend will be more likely to produce some promotion for the quality of completing tasks in crowdsourcing activities. Therefore, we can propose the following definitions:

Definition 4: negative feedback, denoted as F_{nega}, refers to that interest degree will exert negative reaction force on expertise level when it is low enough.

Definition 5: positive feedback, denoted as F_{posi}, refers to that interest degree will exert positive reaction force on expertise level when it is high or not low enough.

Thus, role of reaction force manager here is to assess such reaction force through quantifying two kinds of feedback according to real-time interest degree.

Definition 6: statistical utility is defined as utility reflected by users' historical trend of temporal interest degree.

D. RANKING MANAGER

After processing of reaction force manager, a ranking algorithm based on interest-expertise collaborative awareness is implemented in ranking manager to suggest appropriate user lists for topic-specific tasks as recommendation results. For tasks of topic k, current recommendation result is as the following format:

$$L_k = \{u_4, u_7, u_1, \cdots\} \tag{4}$$

Users in the list is sorted by appropriateness degree, meaning that those with higher appropriateness degree possess higher ranking position.

IV. RECOMMENDATION MECHANISM

In this section, we present mathematical modeling of IntexCrowd for curbing astroturfing problem in crowdsourcing systems. First, Twitter-LDA model is employed to identify topics of newly released tasks as well as tasks in service registry. Then, we use Bayesian posterior inference to estimate users' profile of specific topics. Finally, a ranking algorithm based on interest-expertise collaborative awareness is proposed to suggest user lists for topic-specific tasks.

A. TASK TOPIC MODELING

Latent Dirichlet Allocation (LDA) [7], in which each topic is assumed to be represented by a multinomial distribution of words [6], has been applied to extract topic distribution in different application sceneries [1], [14]. Yet standard LDA may not work well with crowdsourcing tasks because a single task question is short and usually about a single topic. Therefore, in this work, we follow the idea of Twitter-LDA [37] for topic division.

We first assume that each task is associated with a question q. For sake of brevity, we can represent a task as a question q. For topic model, the inputs are Q questions and K topics. For question q, its words are denoted as w_{qm}, in which m ranges from 1 to M_q. The outputs of the topic model are topic labels z_q that represents topic indicator of question q. Let ϕ_k denote the word distribution for topic k, φ' denote the word distribution for background words, and θ denote the topic distribution. Then, let φ denote a Bernoulli distribution that governs the choice between background words and topic words. The generative process of task is as follows. The generation process of questions is illustrated in Fig. 3 and described in **Algorithm 1**.

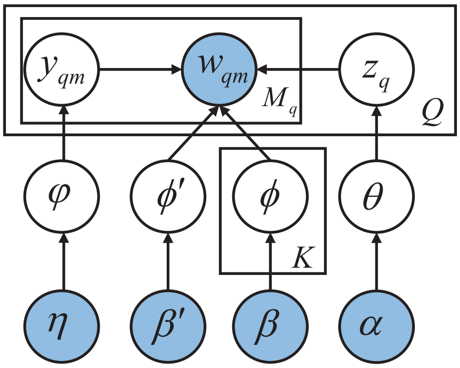


FIGURE 3. Plate for generative process of tasks.

Algorithm 1 The Generation Process of Questions

INPUT: Question set Q
1 : **while** not convergence **do**
2 : Draw $\theta \sim Dir(\alpha)$, $\phi' \sim Dir(\beta)$, $\varphi \sim Dir(\eta)$
3 : **for** each topic $k = 1, 2, \ldots, K$
4 : Draw a word distribution on topic k, $\phi_k \sim Dir(\beta)$
5 : **for** each question $q = 1, 2, \ldots, Q$
6 : Draw a topic $z_q \sim Multi(\theta)$
7 : **for** each word $m = 1, 2, \cdots M_q$
8 : Draw a word category $y_{qm} \sim Bernoulli(\varphi)$
9 : Draw a word $w_{qm} \sim Multi\left(\phi_{z_q}\right)$ if $y_{qm} = 1$, other draw $w_{qm} \sim Multi\left(\phi'\right)$
10 : **end for**
11 : **end for**
12 : **end for**
13 : **end while**
OUTPUT: Topic indicator $k \in [1, K]$

When producing a question, a requester first chooses a topic based on his topic distribution. Then he chooses a bag of words one by one based on the chosen topic or the background model. Each multinomial distribution is governed by some symmetric Dirichlet distribution. We use Gibbs sampling to perform model inference. Due to the space limit we leave out the derivation details and the sampling formulas.

B. USER PROFILING

The objective of user profiling is to calculate users' real-time topical expertise level and topical interest degree.

1) EXPERTISE LEVEL ASSESSMENT

a: MODELING ANSWERS

For each answer of question q provided by user u, it depends on several factors: (1) topic of the question; (2) expertise level of the user on this topic; (3) number of correct answers provided by the user on this topic; (4) the user's instantaneous interest degree for each question at that time point. We assume that expertise level and interest degree have collaborative effect on the quality of completing tasks. Based on the above analysis, we then give the process of generating user u's answer a_{qu} for question q. Intuitively, most users have the ability to provide correct answers for most questions, yet only a few are gurus on the topic. Thus, we assume that users' expertise is drawn from a Gaussian distribution for each topic, i.e. $e_{ku} \sim N\left(\mu_1, \sigma^2\right)$. We assume that there are γ_q different choices $\{1, 2, \cdots \gamma\}$ for each question q. We draw a true answer T_q from a Uniform distribution, a topic indicator $z_q = k$ from Multinomial distribution $Multi(\theta)$ and users' instantaneous interest at the time point he completes the task $i_{z_q u}^{(t)}$ from Gaussian distribution $N\left(\mu_2, \sigma'^2\right)$. Utilizing logistic function, given the topic $z_q = k$, the correct answer given by user u to question q is denoted as a_{qu} which is modeled as follows:

$$p\left(a_{qu} = c | T_q = c, z_q, \pi_{z_q u}, e_{z_q u}, i_{z_q u}^{(t)}\right) = Sig\left(-\pi_{z_q u} e_{z_q u} - i_{z_q u}^{(t)}\right) \quad (5)$$

where $Sig\left(-\pi_{z_q u} e_{z_q u} - i_{z_q u}^{(t)}\right) = \frac{1}{1+\exp\left(-\pi_{z_q u} e_{z_q u} - i_{z_q u}^{(t)}\right)}$ and $\pi_{z_q u}$ is the estimated contribution ratio of user u on topic $z_q = k$.

For all $a_{qu} = c' \neq c$,

$$p\left(a_{qu} = c' | T_q = c, z_q, \pi_{z_q u}, e_{z_q u}, i_{z_q u}^{(t)}\right) = \frac{1 - Sig\left(-\pi_{z_q u} e_{z_q u} - i_{z_q u}^{(t)}\right)}{\gamma_q - 1} \quad (6)$$

Therefore, given the topic $z_q = k$, the joint probability is

$$p\left(a_{qu}, T_q = c, \pi_{z_q u}, e_{z_q u}, i_{qu} | z_q, \mu_1, \mu_2, \sigma^2, \sigma'^2, \gamma_q\right) = p\left(e_{z_q u} | \mu_1, \sigma^2\right) p\left(i_{z_q u}^{(t)} | \mu_2, \sigma'^2\right) p\left(T_q = c | \gamma_q\right) p\left(a_{qu} = c | T_q = c, z_q, \pi_{z_q u}, e_{z_q u}, i_{z_q u}^{(t)}\right) \quad (7)$$

b: HIDDEN VARIABLE INFERENCE

To calculate $e_{z_q u}$ and $i_{z_q u}^{(t)}$ by maximizing the probability of posterior distribution, we formulate the objective function as follows:

$$\begin{aligned} J_{qu} &= -\log p\left(a_{qu}, T_q = c, z_q, \pi_{z_q u}, e_{z_q u}, i_{z_q u}^{(t)} | \mu_1, \mu_2, \sigma^2, \sigma'^2, \gamma_q\right) \\ &= -\log p\left(a_{qu} | T_q = c, z_q, \pi_{z_q u}, e_{z_q u}, i_{z_q u}^{(t)}\right) \\ &\quad -\log p\left(T_q | \gamma_q\right) - \log p\left(e_{z_q u} | \mu_1, \sigma^2\right) - \log p\left(i_{z_q u}^{(t)} | \mu_2, \sigma'^2\right) \\ &\propto -\log p\left(a_{qu} | T_q = c, z_q, \pi_{z_q u}, e_{z_q u}, i_{z_q u}^{(t)}\right) \\ &\quad + \frac{\left(e_{z_q u} - \mu_1\right)^2}{2\sigma^2} + \frac{\left(i_{z_q u}^{(t)} - \mu_2\right)^2}{2\sigma'^2} \end{aligned} \quad (8)$$

Then gradient descent method is used to update e_{z_qu} and $i_{z_qu}^{(t)}$ based on gradients:

$$e_{z_qu}{}^{new} = e_{z_qu}{}^{old} - \lambda \frac{\partial J_{qu}}{\partial e_{z_qu}} \tag{9}$$

$$\left(i_{z_qu}^{(t)}\right)^{new} = \left(i_{z_qu}^{(t)}\right)^{old} - \lambda \frac{\partial J_{qu}}{\partial i_{z_qu}^{(t)}} \tag{10}$$

So far, we obtain user u's topical expertise level as the vector:

$$e_{ku} = [e_{1u}, e_{2u}, e_{3u}, \ldots, e_{Ku}] \tag{11}$$

2) INTEREST DEGREE ASSESSMENT

Through the probabilistic model in Section IV-B.1, we can obtain temporal interest degree as the vector group $\left\{i_{ku}^{(t)}\right\}$ which represents the drifting characteristics of user u's historical interest degree. Next, user u's current interest degree is supposed to be inferred.

Algorithm 2 The Whole Process of User Profiling

INPUT: Topic indicator k, question set Q, user set U, and answers $a_q u$ concerning topic k; parameters: μ_1 ,μ_2,σ^2,σ'^2
1 : **while** not convergence **do**
2 : **for** the question q with topic k
3 : Joint sample $\left(z_q, T_q\right)$ according to Eq.(7)
4 : **for** each user u
5 : Update e_{z_qu} according to Eq.(9)
6 : Update $i_{z_qu}^{(t)}$ according to Eq.(10)
7 : Predict I_{ku} according to Eq.(14)
8 : **end for**
9 : **end for**
10 : **end while**
OUTPUT: e_{ku} according to Eq.(11), I_{ku} according to Eq.(14)

Generally speaking, users' historical tastes may influence his future interests, and more recent interests may have stronger impact on the prediction of future interest degree than the earlier. To imitate the influence of historical behaviors, we apply the exponential decay function [5] to measure drifting:

$$f(t) = \exp\left(-\frac{n-t}{\tau}\right), \quad (t \in \{1, 2, \ldots, n-1\}, \tau > 0) \tag{12}$$

where τ is the kernel parameter, and the value of $n - t$ represents the timestamp between the t-th time instant and current time n. Obviously, the exponential decay function can gradually discount the history of past behavior. Finally, we predict users' current interest degree for different topics as follows:

$$I_{ku} = \sum_{t=1}^{n-1} \exp\left(-\frac{n-t}{\tau}\right) \cdot i_{ku}^{(t)} \tag{13}$$

I_{ku} is a vector as the following format:

$$I_{ku} = [I_{1u}, I_{2u}, \ldots, I_{Ku}] \tag{14}$$

Therefore, the whole user profiling process can be described in **Algorithm 2**.

C. RANKING ALGORITHM

After user profiling, we are supposed to output real-time ranked user lists for tasks of specific topic. And the ranking algorithm incorporates interest-expertise collaborative awareness.

1) REACTION FORCE ASSESSMENT

As aforementioned above, interest degree will exert reaction force on expertise level. We call the reaction force as positive feedback or negative feedback. We set interest threshold as follows:

$$Th_1 = \chi_1 \cdot \mu'$$

If a user's current interest degree of topic k is not lower than Th_1, positive feedback will be exerted on expertise level. Otherwise, negative feedback will be exerted.

When quantifying the feedback, we assume that the higher interest degree is, the stronger positive feedback will be produced. Conversely, the lower interest degree is, the stronger negative feedback will be. Here, we define such feedback with the aid of following fuzzy function:

$$F_{i\to e} = \begin{cases} F_{posi}, & I_{ku} \geq Th_1 \\ F_{nega}, & I_{ku} < Th_1 \end{cases}$$
$$F_{posi} = \frac{1}{1+\exp\{-r\,(I_{ku} - Th_1)\}}$$
$$F_{nega} = \frac{1}{1+\exp\{-r\,(-I_{ku} - Th_1)\}} \tag{15}$$

We further assume that the feedback is not always triggered at any time and that the trigger of it is drawn from a transition probability. The transition probability is related to the frequency or proportion of tasks of such topic that has been participated by this user. Specifically, given topic k, when user u's current interest degree is higher than threshold, there is a probability that positive feedback will be produced. If he has participated in tasks of topic k for many times, he is more likely to treat the task seriously because he pays much attention to it and his current interest degree is relatively high. If he has participated in tasks of topic k for not many times, he is not likely to treat the task seriously because his historical records cannot reflect too much attention. Conversely, when his current interest degree is below the threshold, if he has participated a few tasks of topic k, he may be more likely to treat the task carelessly because his historical records cannot reflect too much attention and his current interest degree is relatively low. On the basis of the above analysis, we can define the transition probability for user u on topic k as follows:

$$p_{ku} = \begin{cases} \dfrac{w_k \cdot n_{ku}}{\sum\limits_{j=1}^{K} w_j \cdot n_{ju}}, & I_{ku} \geq Th_1 \\ 1 - \dfrac{w_k \cdot n_{ku}}{\sum\limits_{j=1}^{k} w_j \cdot n_{ju}}, & I_{ku} < Th_1 \end{cases} \tag{16}$$

where n_{ku} denotes the number of times user u participates in tasks of topic k. In Eq.(16), j ranges from 1 to the total number K. Furthermore, we extend to consider the factor that topics with different prevalence will influence users selecting tasks. And thus we let w_k denote the topic prevalence in the overall service registry during the past period of time. The expression of w_k is as follows:

$$w_k = \frac{N_k}{\sum\limits_{f=1}^{K} N_f} \tag{17}$$

where N_k denotes the number of tasks of topic k in the whole database during the past period of time, and f ranges from 1 to K.

Receiving feedback, we can derive the new expression of topical expertise level as follows:

$$(e_{ku})' = \begin{cases} \left(1 + 0.8 \cdot \left(p_{ku} + F_{posi}\right)\right) \cdot e_{ku}, I_{ku} \geq Th_1 \\ \left(1 + 0.5 \cdot \left(p_{ku} + F_{nega}\right)\right) \cdot e_{ku}, I_{ku} < Th_1 \end{cases} \tag{18}$$

2) STATISTICAL UTILITY ASSESSMENT

As topical interest degrees are usually drifting in the real world, we hold the opinion that more positive statistical trend of interest degree will be more likely to produce some promotion for the quality of completing tasks in crowdsourcing activities.

Here, we call the aforementioned drifting trend as efficiency and introduce two new concepts: abstract efficiency and effective efficiency. As is demonstrated above, there will be some variation between interest degree of each timestamp and of its last timestamp. Absolute value of the variation is called absolute efficiency. If interest degree of a timestamp presents rising state relative to interest degree of its last timestamp, such variation is defined as effective efficiency. Based on the above analysis, we assume that the statistical utility is associated with the proportion of total effective efficiency in total abstract efficiency, and then derive the expression of user u's statistical utility as follows:

$$Uti_{ku} = \frac{\sum\limits_{k} \log\left(1 + V_{ku}^{eff}\right)}{\sum\limits_{k} \log\left(1 + V_{ku}^{abs}\right)} \tag{19}$$

where V_{ku}^{abs} denotes user u's absolute efficiency on topic k and V_{ku}^{eff} denotes effective efficiency.

3) USER RANKING

Given topic k, incompetent users whose expertise level on topic k are too low are filtered firstly. We set the threshold $Th_2 = \chi_2 \cdot \mu$, which means that those whose expertise level on topic k are higher than Th_2 are selected as candidate users $u^{(k)} = \left\{u_1^{(k)}, u_2^{(k)}, u_3^{(k)}, \cdots\right\}$.

After filtering, we formulate a ranking function to calculate ranking scores of candidate users to represent their appropriateness degree, and make candidate users sorted out accordingly. We extend idea of classical method "PageRank" to construct ranking function as follows:

$$S_1 = \lambda \cdot S_e + (1 - \lambda) \cdot S_I \tag{20}$$

where S_e denotes "expertise level" factor and S_I denotes "interest degree" factor.

Next, reaction force is introduced. The feedback interest degree exerts on expertise level is triggered with transition probability p_{ku}. Therefore, we can derive expertise level factor as follows:

$$\begin{aligned} S_e &= (e_{ku})' \\ &= \begin{cases} \left(1 + 0.8 \cdot \left(p_{ku} + F_{posi}\right)\right) \cdot e_{ku}, & I_{ku} \geq Th_1 \\ \left(1 + 0.5 \cdot \left(p_{ku} + F_{nega}\right)\right) \cdot e_{ku}, & I_{ku} < Th_1 \end{cases} \end{aligned}$$

Then, statistical utility is also taken into consideration. The statistical utility is assumed to be viewed as weight exerted on interest degree I_{ku}. We derive interest degree factor as follows:

$$S_I = Uti_{ku} \cdot I_{ku}$$

In all, given topic k, the expression of final ranking score for user u can be formulated in a recursive manner as follows:

$$S_{ku} = \lambda \cdot (e_{ku})' + (1 - \lambda) \cdot Uti_{ku} \cdot I_{ku} \tag{21}$$

where $\lambda \in [0, 1]$ is a damping factor to control the probability of teleportation.

The whole process of generating recommendation results can be described in **Algorithm 3**:

V. CASE STUDY AND EVALUATION

In this section, we will firstly use a case study to illustrate the efficiency of IntexCrowd in case of curbing astroturfing problem for crowdsourcing systems, and then carry out a set of experiments to evaluate the performance of IntexCrowd.

A. CASE STUDY

In this case study, we first consider a building scenario close to reality. Compared with building pure simulation scenario via computer technology, we tend to set up scenario from real-world data. As benchmark dataset of multi-tag labeling crowdsourcing activities whose scene is perfectly matched with ours is never publically available, we utilize another similar real-world dataset named RF as an alternative.

Algorithm 3 The Whole Process of User Ranking

INPUT: e_{ku} according to Eq.(11), I_{ku} according to Eq.(14), and topic indicator k; parameters: Th_1, Th_2, λ
for topic k
  for corresponding users to be selected
   Filter incompetent ones to obtain candidate user set $u^{(k)}$ according to threshold Th_2
    for each user in $u^{(k)}$
     Calculate $F_{i\to e}$ according to Eq.(15)
     Calculate p_{ku} according to Eq.(16)
     Calculate $(e_{ku})'$ according to Eq.(18)
     Calculate Uti_{ku} according to Eq.(19)
     Calculate S_{ku} according to Eq.(21)
     Rank S_{ku} in descending order
    end for
  end for
end for
OUTPUT: Ranking results of users

The RF dataset is extracted from Relevance Finding task of NIST Text REtrieval Conference (TREC) Relevance Feedback Track.[2] Relevance Finding is a 5-tag labeling crowdsourcing task implemented in Amazon Mechanical Turk, where workers of Amazon Mechanical Turk identified the relevance of a given topic and a given document via 5-level rating. The RF dataset includes 98453 judgment records from 753 users for 20232 total examples. 3277 of the examples have prior "gold" labels by NIST. Each record contains the following fields: topic ID, worker ID, gold, and label.

Here, we build the scenario from such a crowdsourcing activity dataset with the following rule: workers are denoted as users in this paper, topic ID is denoted as topic indicator, worker ID is denoted as user identity, gold is denoted as the true answer, and labels are denoted as answers given by users. A whole judgment represents that a user with a worker ID participates in a task of specific topic ID at a timestamp, and gives a label as answer. And we assume that task participation records in the dataset are sorted in chronological order, in order to facilitate the investigation of interest degree dynamics in IntexCrowd.

When a requestor posts a task in crowdsourcing system, a query associated with the task is generated to search for topic indicator. Suppose the returned topic indicator is 20032. Thus, relevant records with the same topic indicator in service registry are sent to user profiling module. Relevant records in RF dataset include 1388 task participation records from 59 users. Given these records, algorithms embedded in user profiling module will infer users' expertise level as well as interest degree on such topic through Gibbs-EM sampling, which is aforementioned in Section IV-B.1. We perform 200 runs of Gibbs-EM sampling. We set Gaussian priors to e_{ku} with mean μ_1 as 36, variance σ^2 as 70. And we set Gaussian priors to instantaneous interest degree with mean μ_2 as 28, variance σ'^2 as 50. After a series of iterative operation, topical expertise level as well as interest degree of the 59 users can be obtained. Due to the limitation of space, we do not provide the calculation details.

Yet not all the users' profile can be pushed to next stage because users without adequate expertise level to guarantee basic requirements need to be filtered. Here, the basic requirements refer to threshold $Th_2 = \chi_2 \cdot \mu$ mentioned above. And we empirically set $\chi_2 = \frac{5}{6}$. If one's expertise level is worse than the threshold, such a user will have a great possibility of not working properly. Table 1 shows expertise level of the 59 users, from which we can see that 20 users meet the expertise level requirement and constitute candidate set. Table 2 illustrates the comprehensive user profile, in which the second and third columns show values of candidate users' expertise level and temporal interest degree respectively. The fourth column in Table 2 shows the values of candidate users' current interest degree obtained according to Eq.(13). Here, parameter τ is set as 3. Variables in Table 1 and Table 2 take the integer.

TABLE 1. Values of users' expertise level concerning topic id 20032.

user id	value	user id	value	user id	value
u100	35	u127	22	u195	34
u101	20	u128	38	u197	24
u102	42	u129	26	u201	24
u103	37	u131	36	u204	21
u105	17	u133	25	u212	43
u106	29	u136	19	u215	38
u108	49	u137	46	u223	20
u109	27	u140	17	u225	22
u110	24	u141	47	u226	52
u112	25	u142	51	u227	18
u113	44	u146	26	u228	28
u115	23	u151	22	u229	43
u116	26	u152	21	u230	21
u117	33	u153	55	u231	20
u118	21	u160	24	u232	29
u119	40	u166	41	u25	24
u120	16	u170	23	u28	21
u121	22	u177	29	u32	29
u124	28	u191	31	u99	24
u126	28	u193	30		

Then, the ranking algorithm can be applied. The whole recommendation is implemented in *MATLAB*. Traditional recommendation methods only consider users' expertise level or interest degree as the key principle, and sometimes cannot function well. In our approach, we further consider the collaborative effect of expertise level and interest degree. To model reaction force that interest degree will exert on expertise level for each candidate user, two variables: feedback $F_{i\to e}$ and transition probability p_{ku}, whose values are shown in the second to the three columns of Table 3, are quantified according to Eq.(15) and (16) respectively. Here, parameters in Eq.(15) are set as $r_1 = 0.3$, $r_2 = 0.15$, $s_1 = 0.65$, $s_2 = 0.45$, and $\chi_1 = \frac{10}{7}$ respectively. To model

[2] http://trec.nist.gov/data/relevance.feedback10.html

TABLE 2. Values of candidate users' profile concerning topic id 20032.

user id	e_{ku}	$i_{ku}^{(t)}$	I_{ku}
u100	35	{25, 31, 28, 37, 26, 25, 20, 29, 19, 29}	42
u102	42	{26, 21, 20, 28, 20, 24, 32, 22, 21, 21}	34
u103	37	{26, 28, 27, 33, 36, 35, 38, 31, 35, 37}	57
u108	49	{20, 23, 26, 22, 24, 21, 25, 24, 26, 23}	38
u113	44	{19, 26, 24, 27, 31, 23, 21, 26, 28, 24}	40
u117	33	{29, 26, 28, 32, 33, 30, 36, 28, 26, 29}	45
u119	40	{36, 35, 36, 35, 33, 30, 29, 27, 28, 25}	41
u128	38	{22, 24, 27, 21, 25, 23, 22, 25, 27, 23}	38
u131	36	{39, 32, 24, 26, 27, 20, 25, 16, 21, 26}	37
u137	46	{28, 32, 33, 27, 25, 29, 30, 26, 24, 27}	42
u141	47	{30, 32, 27, 25, 31, 24, 29, 28, 26, 24}	40
u142	51	{24, 26, 28, 31, 34, 36, 31, 30, 27, 25}	42
u153	55	{26, 29, 31, 32, 35, 30, 27, 28, 25, 27}	42
u166	41	{33, 35, 32, 24, 20, 18, 19, 23, 27, 26}	41
u191	31	{29, 29, 32, 31, 27, 21, 26, 29, 30, 31}	48
u195	34	{24, 29, 30, 27, 32, 36, 31, 28, 26, 29}	45
u212	43	{28, 26, 20, 27, 30, 26, 28, 25, 21, 19}	32
u215	38	{26, 26, 29, 31, 35, 33, 30, 24, 26, 28}	43
u226	52	{28, 30, 35, 33, 31, 27, 26, 27, 29, 25}	41
u229	43	{24, 29, 26, 27, 22, 26, 28, 32, 31, 29}	47

TABLE 3. Values of candidate users' feedback, transition probability, and statistical utility concerning topic id 20032.

user id	$F_{i \to e}$	p_{ku}	Uti_{ku}
u100	0.420	0.073	0.85
u102	0.320	0.104	0.79
u103	0.646	0.128	0.90
u108	0.258	0.087	0.83
u113	0.325	0.079	0.85
u117	0.531	0.093	0.74
u119	0.373	0.064	0.56
u128	0.258	0.051	0.81
u131	0.275	0.088	0.75
u137	0.420	0.076	0.79
u141	0.325	0.091	0.77
u142	0.420	0.067	0.81
u153	0.420	0.135	0.81
u166	0.373	0.084	0.73
u191	0.596	0.093	0.82
u195	0.531	0.085	0.85
u212	0.346	0.068	0.73
u215	0.462	0.071	0.82
u226	0.373	0.089	0.75
u229	0.579	0.117	0.85

statistical utility of temporal interest degree for each candidate user, two variables: V_{ku}^{eff} and V_{ku}^{abs}, are counted firstly. Then, statistical utility Uti_{ku}, whose value is shown in the fourth column of Table 3, is measured according to Eq.(19). Therefore, the final ranking score of each candidate user concerning topic id 20032 can be quantified by Eq.(21). Here, we empirically set the parameter λ in Eq.(21) as 0.65. Fig. 4

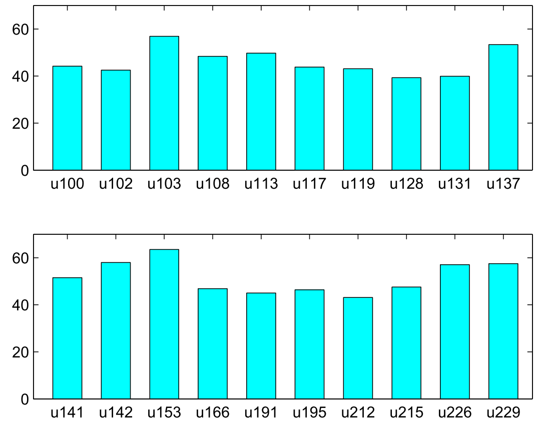


FIGURE 4. The final ranking score of each candidate user.

shows the final ranking score value for 20 candidate users who can be ranked accordingly. Variables $F_{i \to e}$ and p_{ku} are rounded to three decimal places, variable Uti_{ku} is rounded to two decimal places.

From the results, users with identity u153, u142 and u229 are the top 3 users in terms of comprehensive assessment through interest-expertise collaborative awareness, and users with identity u131, u128 and u212 are the worst 3 users.

It is interesting to note that even though some candidate users have relatively high expertise level, their final ranking scores are not as good as others recommended by our recommendation method. We take the user with identity u212 as an example. His expertise level is 43, while his final ranking score is 41.9117. In fact, it is not difficult to understand why this happens. Since the current interest degree of the user is relatively low and is more likely to exert negative reaction force on expertise level, the ranking score will be affected correspondingly. On the other hand, some candidate users who do not possess relatively high expertise level are yet assigned relatively high final ranking score, such as the user with identity u103. This situation obviously indicates the effect of reaction force. Furthermore, it is also interesting to find that even though some users have similar expertise level and current interesting degree, the gap in final ranking score between them is widened, which can be illustrated by the example of users with identity u153 and u226. Difference in historical trend of their temporal interest degree brings about the variation between their statistical utility values, which distinguishes them from a more delicate aspect. As a result, these results exactly verify the rationality of the proposed IntexCrowd, and can help the crowdsourcing platform curb astroturfing by identifying users' characteristics from a more fine-grained perspective.

Going on this case study and given topic ID 20032, we also perform a set of simulation experiments for comparison analysis in terms of curbing astroturfing. Several relevant methods are selected for comparison: **Re-ranking** [3], **IFMF2S** [20], **CRITICAl** [9], and **ZenCrowd** [11]. Details of these methods are summarized in related work.

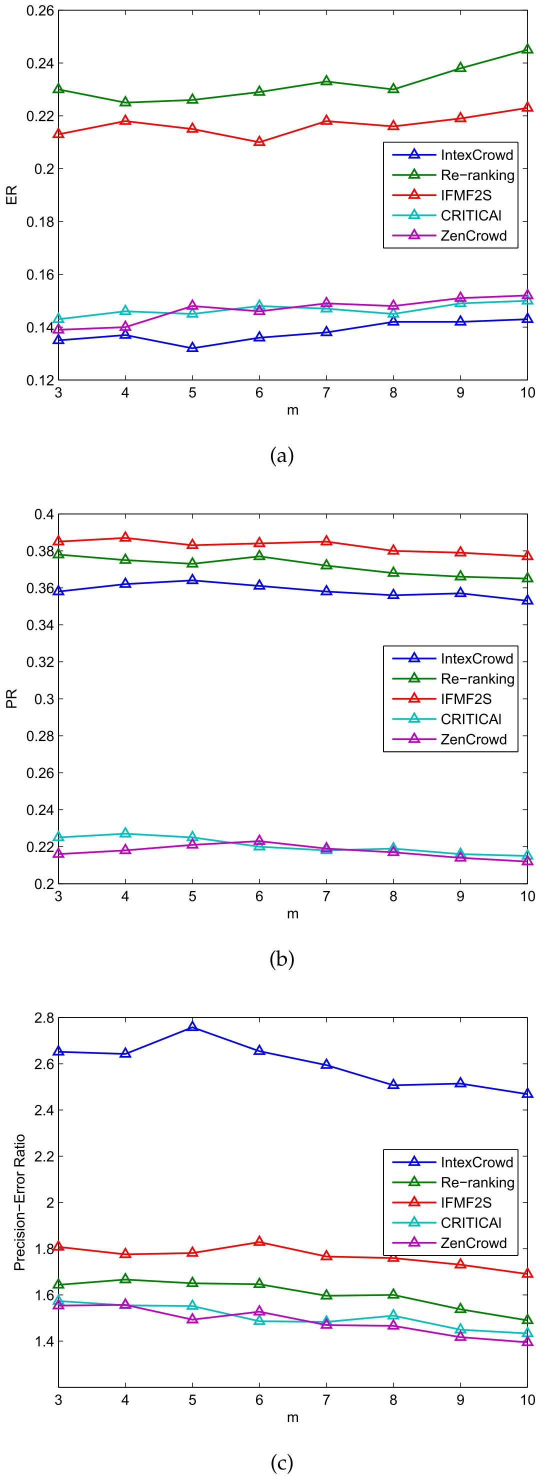


FIGURE 5. Values fluctuation of metrics ER and PR with m changing from 3 to 10. (a) ER. (b) PR. (c) Precision-Error Ratio.

Fig. 5 has three sub-figures. In these figures, the X-axis indicates how many users will be selected after ranking, while the Y-axis represents the values of selected users' performance attributes. Performance attributes include the following three concepts proposed in this study: Error Rate@m (ER@m), Precision Rate@m (PR@m), and Precision-Error ratio@m. Error Rate@m is defined as the number of incorrectly answered questions divided by the total number of questions with respect to top-m recommended users. A lower error rate means that the method is of high quality. For example, given tasks with topic indicator k, the top-3 recommendation result of Re-ranking is Bob, Joy, and Tom. Error rate here refers to the percentage of incorrectly answered questions given by Bob, Joy, and Tom. Precision Rate@m refers to the recommendation accuracy or hit rate of the top-m recommended users. A high precision rate means that the method's recommendation result is more precise. Precision-Error Ratio@m is an evaluation metric that is defined as Precision Rate@m divided by Error Rate@m. Obviously, a higher precision-error ratio value means higher interest degree and expertise level, which reflects better performance. Values of these metrics in Table 4 are rounded to four decimal places.

TABLE 4. Experimental results of IntexCrowd and baselines concerning metrics: ER@5, ER@8, PR@5, PR@8, MAP, MRR.

	ER@5	ER@8	PR@5	PR@8	MAP	MRR
IntexCrowd	**0.1136**	**0.1208**	0.3655	0.3568	**0.4079**	**0.5341**
Re-ranking	0.1950	0.2016	0.3810	**0.3827**	0.3258	0.4385
IFMF2S	0.1854	0.1981	**0.3912**	0.3775	0.3844	0.5016
CRITICAl	0.1247	0.1251	0.2405	0.2529	0.3645	0.4828
ZenCrowd	0.1185	0.1286	0.2132	0.2317	0.3446	0.4617

Fig. 5(a) and 5(b) respectively show the values of selected users' ER@m and PR@m counted from historical records of such topic. And values of selected users' precision-error ratio@m is shown in Fig. 5(c). From Fig. 5(a) and 5(b), we can see that Re-ranking and IFMF2S can select users with not bad precision rate but without ideal error rate, because they only consider users' expertise level and neglect users' interest degree. The proposed IntexCrowd performs not worse than baselines in terms of ER@m and PR@m. From Fig. 5(c), we can see that curve of IntexCrowd is beyond curves of others obviously. All these results reveal the superiority in performance of IntexCrowd which can be attributed as consideration of interest-expertise collaborative awareness.

B. PERFORMANCE EVALUATION

Following case study, we further conduct a set of experiments to evaluate the effectiveness of our mechanism. In this section, we continue to make use of the RF dataset as analog for our evaluation settings. It should be pointed that the series of experiments are on the basis of ready-made division of topic domains. And such setting does not affect experimental validation, because task topic modeling in IntexCrowd directly adopts idea of Twitter-LDA [37].

To evaluate the performance of our method and baselines, besides Error Rate@m and Precision Rate@m, another two evaluation metrics are utilized:

Mean Average Precision: called MAP for short, refers to the mean value of average precisions of all recommendation results.

Mean Reciprocal Rank: called MRR for short, refers to the mean value of reciprocal rank, which is the inverse of the rank of the first relevant user.

Detailed explanations of these measures are in [23]. And baselines for comparison are same with baselines in Section V-A.

With Parameters in Section V-A continuing to be utilized, we implement our method and baselines on training set to suggest top-m lists of users respectively. Incorporating the remaining data, we count the error rate of top-m users recommended by these methods respectively with respect to the whole dataset. Experimental results of the proposed IntexCrowd and baselines are presented in Table 4 and Fig. 6.

The second and the third columns in Table 4 show the values of metrics ER@5 and ER@8 respectively. It is not difficult to find that the proposed IntexCrowd performs well among all the baselines in terms of error rate. Compared with Re-ranking and IFMF2S that only consider users' interest degree, error rate of IntexCrowd is much lower. Compared with others, error rate is still a little lower to different extent. The fourth and the fifth columns of Table 4 show the values of metrics PR@5 and PR@8. And it is not difficult to find that the IntexCrowd works no worse than baselines in terms of recommendation precision. Compared with CRITICAl and ZenCrowd, precision rate of IntexCrowd is much higher because these methods only consider users' expertise level and ignore interest degree. Compared with Re-ranking and IFMF2S, precision rate of IntexCrowd is a little lower. This is because we additionally consider users' expertise level and there are many users who is not knowledgeable about a topic domain but interested in it. Even so, IntexCrowd can archive better performance because it has much lower error rate. The sixth and the seventh columns in Table 4 show the values of metric MAP and MRR respectively. The two metrics express precision of ranking position. It can be seen from the results that IntexCrowd works better than baselines in terms of the two metrics.

Fig. 6 has three subfigures concerning three metrics PR@m, ER@m, and Participation-Error@m Ratio respectively, in which the X-axis indicates the number of top-m users will be selected after ranking, while the Y-axis represents the values of metrics. It can be found from Fig. 6 that ER@m of IntexCrowd is relatively low, PR@m of IntexCrowd is relatively high, and precision-error ratio of IntexCrowd is beyond others. Therefore, IntexCrowd outperforms baselines in terms of ER@m and PR@m.

In all, experimental results shown in this section can well prove that IntexCrowd can curb astrotufing problem better than previous recommendation-based methods. It is more fine-grained than the previous for the following aspects. First, it considers expertise level, which is important in identification of astroturfers. Second, it further considers the time-varying characteristics of interest degree, which is more precise than all the previous methods. Third, it considers statistical utility of drifting interest degrees which can be a latent influence factor. Finally, as it assumes that interest degree and expertise will function together to influence the emergence of astroturfing problem, a ranking function based

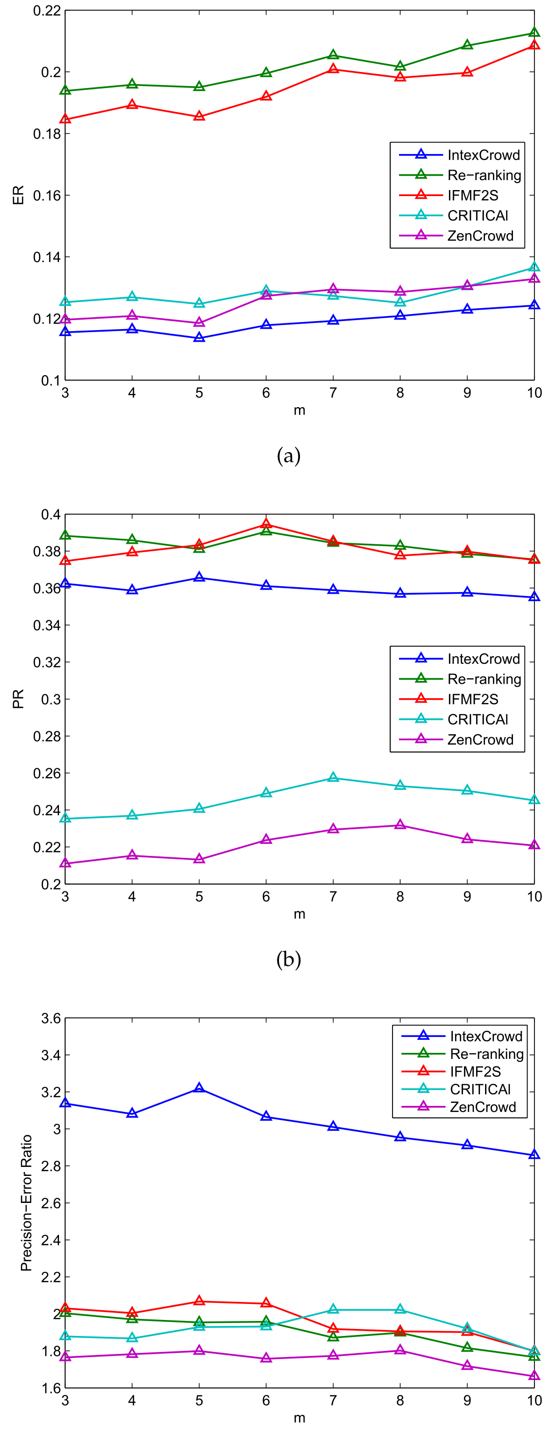


FIGURE 6. Values fluctuation of metrics ER and PR with m changing from 3 to 10. (a) ER. (b) PR. (c) Precision-Error Ratio.

on interest-expertise collaborative awareness is developed to optimize the selection of users.

VI. CONCLUSIONS AND FUTURE WORK

Crowdsourcing is considered as a promising operation mode to apply the wisdoms of crowds to work out large-scale and complex tasks. However, astroturfing problem always hurts the quality of completing tasks and wastes resource. We propose IntexCrowd—fine-grained recommendation mechanism through interest-expertise collaborative awareness to curb astroturfing problem in crowdsourcing systems. Overall, IntexCrowd framework is proposed and the whole flow of IntexCrowd is illustrated firstly. Then, fine-grained recommendation mechanism through interest-expertise collaborative awareness is proposed. Finally, we present a case study and a set of experiments as the demonstration of our approach's efficiency.

In the future, we plan to carry out more similar datasets for massive experiments to verify the applicability of the proposed IntexCrowd mechanism. What's more, we also plan to further research the situation where task recommendation is implemented for a group of users rather than single user, which brings an interesting direction for our future work.

ACKNOWLEDGMENT

The authors would like to thank the guidance of Professor Chaowei Tang, Professor Wenjia Niu, and Professor Hui Tang.

REFERENCES

[1] D. Agarwal and B. Chen, "fLDA: Matrix factorization through latent Dirichlet allocation," in *Proc. 3rd Int. Conf. Web Search Web Data Mining (WSDM)*, New York, NY, USA, Feb. 2010, pp. 91–100.

[2] E. Aldhahri, V. Shandilya, and S. Shiva, "Towards an effective crowdsourcing recommendation system: A survey of the state-of-the-art," in *Proc. IEEE Symp. Service-Oriented Syst. Eng. (SOSE)*, Apr. 2015, pp. 372–377.

[3] V. Ambati, S. Vogel, and J. Carbonell, "Towards task recommendation in micro-task markets," in *Proc. 25th AAAI Conf. Artif. Intell. Workshops*, Jan. 2011, p. 11.

[4] Y. Baba and H. Kashima, "Statistical quality estimation for general crowdsourcing tasks," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2013, pp. 554–562.

[5] H. Bao, Q. Li, S. S. Liao, S. Song, and H. Gao, "A new temporal and social PMF-based method to predict users' interests in microblogging," *Decision Support Syst.*, vol. 55, no. 3, pp. 698–709, 2013.

[6] D. M. Blei, "Probabilistic topic models," *Commun. ACM*, vol. 55, no. 4, pp. 77–84, Apr. 2012.

[7] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.

[8] M. Bouguessa, B. Dumoulin, and S. Wang, "Identifying authoritative actors in question-answering forums: The case of Yahoo! answers," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 866–874.

[9] I. Boutsis and V. Kalogeraki, "On task assignment for real-time reliable crowdsourcing," in *Proc. IEEE 34th Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2014, pp. 1–10.

[10] C. C. Cao, J. She, Y. Tong, and L. Chen, "Whom to ask?: Jury selection for decision making tasks on micro-blog services," *Proc. VLDB Endowment*, vol. 5, no. 11, pp. 1495–1506, 2012.

[11] G. Demartini, D. E. Difallah, and P. Cudré-Mauroux, "Zencrowd: Leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking," in *Proc. 21st World Wide Web Conf.*, 2012, pp. 469–478.

[12] D. E. Difallah, G. Demartini, and P. Cudré-Mauroux, "Pick-a-crowd: Tell me what you like, and i'll tell you what to do," in *Proc. 22nd Int. Conf. World Wide Web*, 2013, pp. 367–374.

[13] J. Guo, S. Xu, S. Bao, and Y. Yu, "Tapping on the potential of q&a community by recommending answer providers," in *Proc. 17th ACM Conf. Inf. Knowl. Manage.*, 2008, pp. 921–930.

[14] M. Harvey, F. Crestani, and M. J. Carman, "Building user profiles from topic models for personalised search," in *Proc. 22nd ACM Int. Conf. Inf. Knowl. Manage. (CIKM)*, San Francisco, CA, USA, Oct. 2013, pp. 2309–2314.

[15] C. Ho, S. Jabbari, and J. W. Vaughan, "Adaptive task assignment for crowdsourced classification," in *Proc. 30th Int. Conf. Mach. Learn.*, 2013, pp. 534–542.

[16] C. Ho and J. W. Vaughan, "Online task assignment in crowdsourcing markets," in *Proc. AAAI*, vol. 12. 2012, pp. 45–51.

[17] J. Howe, "The rise of crowdsourcing," *Wired Mag.*, vol. 14, no. 6, pp. 1–4, Jun. 2006.

[18] P. Jurczyk and E. Agichtein, "Discovering authorities in question answer communities by using link analysis," in *Proc. 16th ACM Conf. Conf. Inf. Knowl. Manage.*, 2007, pp. 919–922.

[19] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *J. Assoc. Comput. Mach.*, vol. 46, no. 5, p. 604, 1999.

[20] C. Lin, E. Kamar, and E. Horvitz, "Signals in the silence: Models of implicit feedback in a recommendation system for crowdsourcing," in *Proc. 28th AAAI Conf. Artif. Intell.*, 2014, pp. 908–914.

[21] X. Liu, W. B. Croft, and M. B. Koll, "Finding experts in community-based question-answering services," in *Proc. ACM CIKM Int. Conf. Inf. Knowl. Manage.*, Bremen, Germany, Oct. 2005, pp. 315–316.

[22] F. Ma *et al.*, "Faitcrowd: Fine grained truth discovery for crowdsourced data aggregation," in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2015, pp. 745–754.

[23] C. D. Manning, H. Schütze, and P. Raghavan, *Introduction to Information Retrieval*. Cambridge, U.K.: Cambridge Univ. Press, 2008.

[24] H. Rahman, S. B. Roy, S. Thirumuruganathan, S. Amer-Yahia, and G. Das, "Task assignment optimization in collaborative crowdsourcing," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2015, pp. 949–954.

[25] V. Rakesh, W. C. Lee, and C. K. Reddy, "Probabilistic group recommendation model for crowdfunding domains," in *Proc. 9th ACM Int. Conf. Web Search Data Mining*, 2016, pp. 257–266.

[26] V. C. Raykar and S. Yu, "Eliminating spammers and ranking annotators for crowdsourced labeling tasks," *J. Mach. Learn. Res.*, vol. 13, pp. 491–518, Feb. 2012.

[27] V. S. Sheng, F. Provost, and P. G. Ipeirotis, "Get another label? Improving data quality and data mining using multiple, noisy labelers," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 614–622.

[28] Y. Tian and J. Zhu, "Learning from crowds in the presence of schools of thought," in *Proc. 18th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2012, pp. 226–234.

[29] L. Tran-Thanh, T. D. Huynh, A. Rosenfeld, S. D. Ramchurn, and N. R. Jennings, "BudgetFix: Budget limited crowdsourcing for interdependent task allocation with quality guarantees," in *Proc. Int. Conf. Auto. Agents Multi-Agent Syst.*, 2014, pp. 477–484.

[30] F. Xu, Z. Ji, and B. Wang, "Dual role model for question recommendation in community question answering," in *Proc. 35th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2012, pp. 771–780.

[31] L. Yang *et al.*, "Cqarank: Jointly model topics and expertise in community question answering," in *Proc. 22nd ACM Int. Conf. Inf. Knowl. Manage.*, 2013, pp. 99–108.

[32] J. Yi, Q. Zhuge, J. Hu, S. Gu, M. Qin, and E. H. Sha, "Reliability-guaranteed task assignment and scheduling for heterogeneous multiprocessors considering timing constraint," *J. Signal Process. Syst.*, vol. 81, no. 3, pp. 359–375, 2015.

[33] M. C. Yuen, I. King, and K. S. Leung, "TaskRec: Probabilistic matrix factorization in task recommendation in crowdsourcing systems," in *Proc. Int. Conf. Neural Inf. Process.*, 2012, pp. 516–525.

[34] J. Zhang, M. S. Ackerman, and L. Adamic, "Expertise networks in online communities: Structure and algorithms," in *Proc. 16th Int. Conf. World Wide Web*, 2007, pp. 221–230.

[35] W. Zhang and J. Wang, "A collective Bayesian poisson factorization model for cold-start local event recommendation," in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2015, pp. 1455–1464.

[36] T. Zhao, N. Bian, C. Li, and M. Li, "Topic-level expert modeling in community question answering," in *Proc. SDM*, vol. 13. 2013, pp. 776–784.
[37] W. X. Zhao *et al.*, "Comparing twitter and traditional media using topic models," in *Proc. 33rd Eur. Conf. IR Res. Adv. Inf. Retrieval*, Dublin, Ireland, Apr. 2011, pp. 338–349.
[38] Z. Zhao, F. Wei, M. Zhou, W. Chen, and W. Ng, "Crowd-selection query processing in crowdsourcing databases: A task-driven approach," in *Proc. Int. Conf. Extending Database Technol.*, 2015, pp. 397–408.
[39] Z. Zhao, D. Yan, W. Ng, and S. Gao, "A transfer learning based framework of crowd-selection on twitter," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2013, pp. 1514–1517.
[40] Y. Zheng, J. Wang, G. Li, R. Cheng, and J. Feng, "QASCA: A quality-aware task assignment system for crowdsourcing applications," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2015, pp. 1031–1046.
[41] G. Zhou, S. Lai, K. Liu, and J. Zhao, "Topic-sensitive probabilistic model for expert finding in question answer communities," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage.*, 2012, pp. 1662–1666.

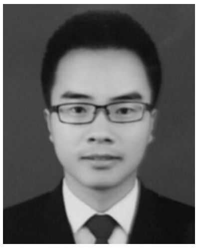
ZHIWEI GUO received the B.S. degree from Zhengzhou University in 2013. He is currently pursuing the Ph.D. degree with Chongqing University. Since 2014, he has been involved in data mining and machine learning. His research interests include contain recommender system and neutral networks.

CHAOWEI TANG received the B.S., M.S., and Ph.D. degrees in communication and information system from Chongqing University in 1987, 1990, and 1993, respectively. He is currently a Professor with the College of Communication engineering, Chongqing University. His research interests include intelligent information processing, computer vision, data mining, and Internet of things.

WENJIA NIU received the B.S. degree from Beijing Jiaotong University in 2005, and the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences. Since 2017, he has been appraised as a Professor and the Doctoral Supervisor with Beijing Jiaotong University. His research interests include computer networks, pattern recognition, data mining, and network security.

YUNQING FU received the B.S., M.S., and Ph.D. degrees in computer software and theory from Chongqing University in 1992, 1995, and 2001, respectively. He is currently a Professor with the College of Software, Chongqing University. His research interests include intelligent information processing, computer networks, and Internet of things.

TONG WU received the B.S. degree from Qingdao University in 2002, and the Ph.D. degree from the People's Public Security University of China. He is currently an Engineer with Tsinghua University. Since 2012, he has been a Judicial Authenticator with the Digital Evidence Judicial Identification Center, Beijing Association of Forensic Science. His research interests include cybercrime, digital forensics, and network security.



HAIYANG XIA received the B.S degree from Xi'an Shiyou University in 2015, where he is currently pursuing the master's degree. His research interests include social network privacy, text mining, data mining, and computer vision.

HUI TANG received the B.S. degree from Lanzhou University in 1992, the M.S. degree from the Institute of Computing Technology, Chinese Academy of Sciences, in 1995, and the Ph.D. degree from the Institute of Acoustics, Chinese Academy of Sciences in 1998. Since 2004, he has been the Founding Director of the High Performance Network Laboratory, Institute of Acoustics. He is currently an Adjunct Professor with Chongqing University. His research interests include the next-generation broadband wireless mobile network.