

供需匹配中的非诚信行为预防

张少杰 鹿旭东 郭伟 王世鹏 何伟

山东大学软件学院 济南 250000

(sagechang2020@outlook.com)

摘要 供需匹配问题可以通过社交网络(Social Network,SN)下的众包模式得到解决。但由于实际应用中的非合作约束,以及社交网络的隐私保护机制,众包的参与者具有通过非诚信行为获利的动机与条件。这类行为会影响公平性原则,并将导致网络中信任链的崩塌,最终使得整个众包模式的供需匹配规则失效。为解决众包供需匹配方法中的非诚信问题,考虑通过分布式公开记账的方式来确保成员如实汇报个体的行为与状态,并通过核对公开的信息来寻找两类非诚信者。此外,设计基于信誉的惩罚机制来对抗非诚信行为,并最终通过理论分析证明了此机制的有效性与可行性。在此机制下,众包参与者的最优策略便是保证诚实。

关键词: 供需匹配;众包;非诚信行为;非合作;社交网络

中图法分类号 TP399

Prevention of Dishonest Behavior in Supply-Demand Matching

ZHANG Shao-jie, LU Xu-dong, GUO Wei, WANG Shi-peng and HE Wei

School of Software, Shandong University, Jinan 250000, China

Abstract Supply-demand matching problem can be solved by crowdsourcing in social networks (SN). However, due to the non-cooperative constraints in practical applications and the privacy protection mechanism of social networks, participants of crowdsourcing have the motivation and conditions to profit from dishonest behaviors. This kind of behavior affects the fairness principle, and will lead to the collapse of the trust chain in the network. In order to solve the problem of dishonest behavior in the crowdsourcing supply-demand matching method, this paper considers the distributed public accounting to ensure that members truthfully report individual behavior and status, and looks for two types of dishonest individuals by checking the public information. This paper also designs a punishment mechanism based on reputation to counter dishonest behavior. Finally, the validity and feasibility of our mechanism are proved by theoretical analysis. Under the mechanism, the best strategy for crowdsourcing participants is to be honest.

Keywords Supply-demand matching, Crowdsourcing, Dishonest behavior, Non-cooperative, Social networks

1 引言

众包作为一种灵活的服务计算范式,使得无处不在的人力资源可以按需求提供给请求者^[1]。因此,众包常常应用于专业团队组建问题^[2-8],此类问题的目标是寻找一个能覆盖任务全部所需技能的团队^[9]。与之类似,众包也可用于解决供需匹配问题,这类问题主要出现在产品的协调化生产中,在该场景下,需求方生产一批产品需要多家供应商协同完成。在复杂的供需匹配问题中,任务的请求者希望以更低的成本构建一个在种类和数量上同时覆盖任务需求的供应网,而任务的参与者则希望在不改变自身薪酬水平的情况下尽可能获得更多的订单。为达到以上目的,以多主体系统的视角和方法

来理解众包系统,可以将众包和多主体系统的研究联系起来^[10]。在非合作式社交网络下的众包模式供需匹配方法中,任务请求者利用成员邻居的部分信息拓展供应网,以新的低成本节点替换旧的高成本节点,完成对任务需求的覆盖^[3,11-12]。该方法还允许在SN中的各任务请求者之间进行协调来交换资源,以实现SN总收益的提升。考虑到个体的自私性与网络的隐私保护,该方案无法保证SN中的个体如实汇报行为与状态等信息,因此存在因成员非诚信行为导致供需匹配规则失效的风险。

尽管目前已有针对众包模式的诚信机制的研究^[13-14],但我们发现了其中的一些片面性与不足:1)这些研究主要关注队伍组建过程中谈判双方的真实定价问题,以及任务参与者

到稿日期:2020-09-10 返修日期:2020-10-11

基金项目:国家重点研发计划(2019YFB1705904);山东省重大科技创新工程项目(2019JZZY020505,2019JZZY010109,2018YFJH0506)

This work was supported by the National Key Research and Development Project of China(2019YFB1705904) and Science and Technology Development Plan Project of Shandong Province(2019JZZY020505,2019JZZY010109,2018YFJH0506).

通信作者:鹿旭东(dongxul@sdu.edu.cn)

对自身工作成本、效率等信息的谎报,但缺少针对众包参与者隐瞒或谎报自身行为、状态以及社交关系等情形的研究;2)这些研究主要站在任务请求者的角度,考虑如何防止任务参与者的非诚信行为,从而提高自身的收益,但缺乏针对任务请求者的非诚信行为的研究;3)这些研究主要应用于众包模式下的专业协作团队组建方法,缺乏针对众包供需匹配方法特有的非诚信行为的检查机制。

在此背景下,本文重点关注众包模式下供需匹配方法中的非诚信行为,我们首次意识到了该方法中可能存在两类非诚信行为,这也是本文的主要贡献:1)恶意隐瞒行为,高成本节点向请求者隐瞒自己的低成本邻居以防止自己的订单被取消;2)立约者扣押行为,请求者发布虚假任务或刻意延长供应网构建时间,锁死对立约者的雇佣权,在与其它请求者协调的过程中索取献金以获得收益。为此,我们设计了一套基于分布式记账的非诚信行为检查与惩罚机制,包含:1)在对 SN 的长期维护过程中,由完成供应网构建的请求者检查以往 SN 中的公开信息以及供应者的历史恶意隐瞒行为;2)在任务供应网构建过程中,由 SN 中的其他节点持续监督请求者的行为与状态信息,检查请求者对立约者的扣押行为;3)建立基于信誉的惩罚机制,通过提高非诚信者与 SN 中其他节点的合作信任代价来惩罚非诚信者。

理论分析表明,本文设计的机制通过分布式公开记账的手段,确保了 SN 成员提供信息的真实性。由于该机制符合诚信者的利益,诚信者在 SN 中具有充分的动机来如实汇报信息和监督检查异常的嫌疑节点。同时,对公开信息的检查保证了成员的非诚信行为可以被发现。我们还从博弈的角度对机制的有效性进行了分析,证明了在本文的机制下,SN 中成员的最优策略便是保证诚实。

2 相关研究

非合作网络中的自私个体存在不诚信的动机,且不诚信的个体将不利于诚信者的利益^[15]。以往对于众包模式下的非诚信行为的研究主要集中于个体的真实定价问题^[13-14]与保障服务质量的问题^[16-17]。Liu 等研究了众包问题中的诚信定价问题,提出了基于反向拍卖的 VCG 机制和 TruTeam 机制等 4 个真实定价机制^[13]。其中,最典型的 VCG 机制的报酬计算依据的是边界价格理论,支付的报酬是临界值。该临界值的计算方法是将计算报酬的成员从工作池中剔除后,按照算法重新选一次参与者并得到他们的总报酬,用该总报酬减去该成员参与时的总报酬,再减去该成员的报价即可。但由于 VCG 机制是 NP 困难的,相较之下 TruTeam 机制最符合应用的实际。Wang 等则提出了基于二叉树转化和贪婪方法两种诚信机制^[14-18],并表明在小规模应用中提取二叉树的方法的效果更好,但在大规模众包应用中贪婪方法的表现更佳。Tran-Thanh 等研究了在预算有限的前提下如何雇佣更高质量专家的问题^[16],针对该问题设计了经典的多臂老虎机(Multi-Armed Bandit, MAB)模型。在该模型中,请求者花费其总预算 B 的第一个 ϵB 预算来获得工人的质量估计,并花费剩余的 $(1-\epsilon)B$ 预算来最大化基于这些估计的任务质量。Tran-Thanh 还提出了一种 BudgetFlx 算法^[17],该算法确

定了相互依赖的任务数量,以及在给定预算限制的情况下为每个子任务支付的价格。此外,BudgetFlx 还为 workflow 中每个阶段的输出的准确性提供了质量保证。

与上文针对真实定价问题与服务质量问题的诚信机制不同,本文提出了新的针对 SN 隐私性带来的邻居信息隐瞒行为和扣押立约者行为的非诚信行为惩罚机制,采用分布式记账的方法,利用群体信息的公开来对抗非诚信行为。

3 问题描述

3.1 供需匹配问题

首先,给出符号的定义,如表 1 所列。

表 1 符号定义
Table 1 Definition of notation

Notation	Definition
a_i	代理 i
$G(a_i)$	a_i 拥有的产品供应能力
$M(a_i)$	a_i 的各产品供应能力上限
$C(a_i)$	a_i 的各产品单位供应成本
t	任务 t
$G(t)$	任务 t 所需的产品供应种类
$R(t)$	任务 t 对各产品的需求量
$E(t)$	任务 t 能为 I_t 带来的价值
N_t	任务 t 的供应网
$O(t)$	N_t 中包含的订单的详情
$p(a_i, g_j, t)$	订单中 a_i 为任务 t 供应产品 g_j 的薪酬
$U_s(t)$	任务 t 尚未被完全满足的需求
$N(a_i)$	a_i 的直接邻居
g_j	产品需求 j
$\max(a_i, g_j)$	a_i 当前的供应 g_j 的能力上限
$c(a_i, g_j)$	a_i 供应 g_j 的单位成本
I_t	任务 t 的发起者
$Pro(t)$	任务 t 为 I_t 带来的纯收益(利润)
$r(t, g_j)$	任务 t 对产品 g_j 的需求量
$e(t, g_j)$	任务 t 中 g_j 带来的价值
Ω_t	供应网 N_t 已包含的代理
$q(a_i, g_j, t)$	订单中 a_i 为任务 t 供应产品 g_j 的量
$\eta(I_t, a_i)$	I_t 与 a_i 间额外的沟通代价系数
$q_{uc}(t, g_j)$	t 中未被满足的产品 g_j 的量

公开记社交网络 $SN=\langle A,E\rangle$ 是一个无向不带权图,其中 $A=\{a_1,a_2,\cdots,a_m\}$ 是图中所有成员节点的集合, $\forall (a_i,a_j)\in E$ 表示节点 a_i 和 a_j 之间存在社交联系。 $\forall a_i\in A$ 由四元组 $\langle G(a_i),M(a_i),C(a_i),N(a_i)\rangle$ 定义。其中, $G(a_i)=\{g_1,g_2,\cdots\}$ 代表 a_i 拥有的产品供应能力的种类; $M(a_i)=\{\max(a_i,g_1),\cdots,\max(a_i,g_{|G(a_i)|})\}$ 表示某段时间内 a_i 供应 $g_j\in G(a_i)$ 的最大上限量; $C(a_i)=\{c(a_i,g_1),\cdots,c(a_i,g_{|G(a_i)|})\}$ 表示 a_i 供应单位量 $g_j\in G(a_i)$ 的成本; $N(a_i)$ 是 a_i 的直接邻居,即 $N(a_i)=\{a_j|\langle a_i,a_j\rangle\in E\}$ 。

任务 t 由四元组 $\langle I_t,G(t),R(t),E(t)\rangle$ 定义。其中, I_t 是任务请求者; $G(t)=\{g_1,g_2,\cdots\}$ 表示完成任务 t 所需的全部产品供应的种类; $R(t)=\{r(t,g_1),\cdots,r(t,g_{|G(t)|})\}$ 表示任务 t 对各种产品供应的需求量; $E(t)=\sum_{j=1,\cdots,|G(t)|}e(t,g_j)$ 表示任务 t 中各产品供应能为 I_t 带来的价值(非利润)总和。

针对任务 t 的供应网 N_t 由 $\langle t,\Omega_t,O(t),U_s(t)\rangle$ 定义。其中, Ω_t 为 N_t 已雇成员集合; $O(t)=\{(t,a_i,g_j,q(a_i,g_j,t),p(a_i,g_j,t),\eta(I_t,a_i)),\cdots,(t,a_p,g_q,q(a_p,g_q,t),p(a_p,g_q,t),\eta(I_t,a_p))\}$ 表示 N_t 中包含的订单的详情集合,其中 $q(a_i,g_j,$

t)表示订单中 a_i 为任务 t 供应产品 g_j 的量, $p(a_i, g_j, t) = q(a_i, g_j, t) * c(a_i, g_j)$ 表示该订单的金额也就是 a_i 的薪酬, $\eta(I_t, a_i)$ 是 I_t 与 a_i 间额外的沟通代价系数,由社交关系影响的初始沟通代价系数 $\eta'(I_t, a_i)$ 与诚信机制影响的不信任系数 $Distrust(a_i)$ 求和得到; $Us(t) = \{us(t, g_j, q_{us}(t, g_j)), \dots, us(t, g_q, q_{us}(t, g_q))\}$ 表示任务 t 尚未被完全满足的需求,其中 $q_{us}(t, g_j) = r(t, g_j) - \sum_{(t, a_i, g_j, \dots) \in O(t)} q(a_i, g_j, t)$ 表示 t 中未被满足的 g_j 需求量。

供需匹配问题指,给定 $SN = \langle A, E \rangle$ 与 t , 任务发起人 I_t 希望构建一个在需求的种类和数量上同时满足要求的供应网 $N_t = \langle \Omega_t, O(t), \emptyset \rangle$, 使得:1) 任务 t 在需求种类和数量上都被 N_t 完全覆盖;2) N_t 不包含冗余成员,每个成员至少提供一种产品的供应;3) 尽可能减小成本,即提高收益 $Pro(t) = E(t) - \sum_{(t, a_i, g_j, \dots) \in O(t)} p(a_i, g_j, t) * (1 + \eta(I_t, a_i)) * c(a_i, g_j)$ 。

3.2 供需匹配问题

本节首先定义几个角色概念以及对应的产能概念。

定义 1(自由者、立约者、供应者) 对于某种产品 g_j , 若某时刻拥有 g_j 供应能力的节点 a_i 不拥有 g_j 订单, 此时 a_i 被称为自由者; 若 a_i 拥有 g_j 订单, 但由于所在 N_t 尚未构建完成, g_j 供应工作尚未开始, 此时 a_i 被称为立约者; 供应开始后直到结束前, a_i 被称为供应者。

定义 2(空闲产能、上锁产能、禁用产能) 对于产品 g_j 的自由者, 拥有的 g_j 产能称为空闲产能, 可由自由者自行决定是否向请求者提供供应; 立约者的 g_j 产能为上锁产能, 受合约限制不可由立约者随意支配; 供应者的 g_j 产能为禁用产能, 已开始为雇主提供供应, 受合约限制无论如何不可提供给其他请求者。

本文众包模式的供需匹配方案包含两种主要算法, 即供应网构建算法和协调算法。其中, 向供应网构建算法输入 SN 和任务 t 的需求, 输出初步覆盖需求的符合 I_t 利益的供应网 N_t , 经协调算法, 输出优化后的供应网。两种算法的关系如图 1 所示。供应网构建算法如算法 1 所示。

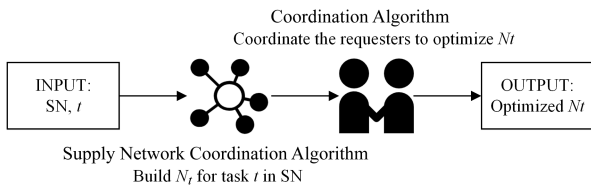


图 1 两种算法的关系

Fig. 1 Relationship of two algorithms

算法 1 供应网构建算法

1. initialize: $\Omega_t = I_t, \forall g_j \in G(t), q_{us}(t, g_j) = r(t, g_j), O(t) = \emptyset$
2. initialize: $N_t = \langle \Omega_t, O(t), Us(t) \rangle$
3. for each $a_i \in \Omega_t$
4. for $a_x \in N(a_i) \cup a_i$
5. decide($N(t), a_x$)
6. update $N(t)$ and a_x
7. if $\forall g_j \in G(t), q_{us}(t, g_j) = 0$
8. terminate this Algorithm
9. end for
10. end for

步骤 1、步骤 2 初始化 $N(t)$ 的成员只包含 I_t , 不含任何订单, 剩余未满足需求等于 t 完整的初始需求。遍历供应网 N_t 的每个已包含成员与其每个邻居节点(见步骤 3、步骤 4), 对其执行 decide 决策算法(见步骤 5), 并根据决策算法返回的信息更新供应网 N_t (见步骤 8), 建立新订单或者以低成本的新订单替换高成本的旧订单, 直至 t 的全部需求被满足(见步骤 7、步骤 8)。

算法 1 中步骤 5 中的决策方法基于分布式谈判, 用于决定是否及如何建立某订单, 包含“ I_t 向 a_i 发出订单 offer”“ a_i 根据自身产能状态对订单做出 response”“双方做出最后 confirm”3 个阶段。谈判细节与本文将讨论的非诚信行为关系不大, 这里不做详细描述。但需要注意: 1) 作为后文协调算法的依据, 在 decide 方法中, 当 a_i 的产能状态为上锁产能时, a_i 将回应 mark 以声明“假若自己没有 I_t 雇佣, 原本可接受 I_t 订单的量”, 而收到 mark 回应的 I_t 只得暂时放弃对 a_i 的雇佣, 继续步骤 4 的下一轮循环; 2) 由于供应者在一段时间内能提供的产能是有限的, 一个供应者不能接受超过其当前剩余产能 $\max(a_i, g_j)$ 的量的订单。

供应网构建算法后可以得到初步覆盖任务需求的供应网。但若存在 mark, 则此时 N_t 很可能还不是最优的。这是由于 I_t 在构建 N_t 过程中, 收到了 a_i 的回应 $mark(t, a_i, g_j, Q_{temp}, P_{temp}, t_2)$, 此时 I_t 只得暂时放弃 a_i 。 I_t 构建完 N_t 之后发现, 若雇佣 a_i 供应 g_j , 可以相比当前 N_t 节约成本 C_1 ; 而对于立约者 a_i 的雇佣者 I_{t_2} , 若除 a_i 外有其他备选供应者可替代完成这部分 g_j 供应, 且替代后升高的成本 $C_2 < C_1$, 则 I_t 与 I_{t_2} 的总成本存在优化的余地。

协调算法通过调整对立约者 a_i 的雇佣权, 来同时优化协调双方的收益, 如算法 2 所示。

算法 2 协调驱动算法

1. for each $mark(t, a_i, g_j, Q_{temp}, P_{temp}, t_2)$ that I_t received
2. Cooperate($I_t, I_{t_2}, a_i, g_j, Q_{temp}, P_{temp}$)
3. end for

循环遍历 I_t 收到的每个 mark 回应, 并对该回应涉及的 I_{t_2} 进行关于 (a_i, g_j) 的协调。协调方式为: 1) I_t 评估, 若以 a_i 的 g_j 订单更新当前的 N_t , 是否能够节约成本, 若能则计算节约的成本 C_1 ; 2) I_{t_2} 检查是否有其他节点可替代 a_i 供应 g_j , 若有则计算因此损失的成本 C_2 ; 3) 当 $C_1 > C_2$ 时, I_t 向 I_{t_2} 提供介于 $[C_2, C_1)$ 的献金, 双方完成雇佣权的交换。这样就同时提高了双方利益, 同时也不违反立约者 a_i 的利益。

3.3 恶意隐瞒问题

在本文描述的问题中, 恶意隐瞒行为的发生是出于 SN 的隐私性和非合作性两方面原因。

(1) 由于 SN 中的隐私保护, 请求者只能访问其直连的社交邻居的有限信息, 不允许获取包括参与者社交关系在内的一些隐私信息, I_t 需经过供应网 N_t 中立约者 a_i 的推荐才能获知非直连邻居 a_j 的存在。而任务请求者无法得知供应网成员是否真实地汇报了其邻居节点信息。

(2) 由于非合作性的基本原则, 请求者与供应者双方都希望最大化自身利益。请求者希望雇佣尽可能廉价的供应者来完成订单需求, 而供应者则希望在不降低其供应成本(也就是单位工作量薪酬)的情况下, 尽可能地增加自己的订单量。

恶意隐瞒问题指,请求者 I_i 在供应网 N_i 构建过程中雇佣 a_i 来供应产品 g_j , a_i 拥有直连邻居 a_j , 而 a_j 同样拥有 g_j 供应能力且成本更低,此时 a_i 向 I_i 如实汇报 a_j 会有自己承担的订单被取消(被 a_j 代替)的风险,基于非合作原则下个体的自私性, a_i 倾向于向 I_i 隐瞒邻居 a_j 的存在,如图 2 所示。这种行为损害了 I_i 和 a_j 的利益,同时也违反了公平竞争原则。

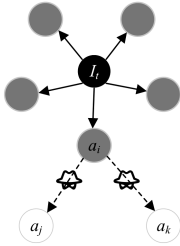


图 2 恶意隐瞒
Fig. 2 Vicious hiding

3.4 立约者扣押问题

在众包模式的供需匹配方法中,社交资源丰富的请求者有更多的机会选择低成本节点,更容易构建低成本的供应网。而协调算法则允许请求者之间通过献金来交换立约者的雇佣权,这使得社交资源丰富的请求者有机会获得额外的献金收益,但也导致了立约者扣押问题发生的可能。

立约者扣押问题指,SN 中的个体 I_i 发布虚假的任务 t , 或刻意延长 N_i 的构建过程,使得 N_i 中的立约者长期处于被锁死的状态,无法与 SN 中的其他请求者达成订单,而 I_i 则向其他请求者索取献金以交换对所扣押的立约者的雇佣权,如图 3 所示。该行为是不正当得利行为,会使其他请求者多支出一笔献金,也可能导致 N_i 中的立约者损失订单。

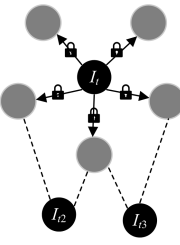


图 3 扣押立约者
Fig. 3 Detaining contractors

4 非诚信行为检查与惩罚机制

4.1 恶意隐瞒行为检查机制

解决恶意隐瞒问题的关键在于,高成本个体倾向于隐瞒低成本邻居,而低成本个体则没有隐瞒高成本邻居的动机,因此我们可以通过检查 SN 中不对称的节点邻居关系来对抗恶意隐瞒行为。

本文通过分布式公开记账的方式,在对 SN 的长期维护过程中发现不对称的邻居关系。

4.1.1 公开记账

I_i 在执行供应网构建算法时记录 N_i 的构建过程,记账规则如下。

(1)算法 1 中的步骤 3(for each $a_i \in \Omega_i$)中,每访问一个新

节点, I_i 向整个 SN 公开记账,即 $visit(t, a_i, I_i)$,表明 N_i 构建过程中已要求 a_i 提供其邻居信息,记录发送者是 I_i 。

(2)算法 1 中的步骤 4(for $a_x \in N(a_i) \cup a_i$)中,每访问一个 a_i 的邻居, I_i 向 SN 公开记账: $relation(t, a_i -> a_x, I_i)$,表明 a_i 向 I_i 推荐了邻居 a_x ,记录发送者是 I_i 。

(3)每当算法 1,即供应网构建算法完成时, I_i 向整个 SN 公开算法中 N_i 雇佣的最后一个节点 a_x 及其推荐人 a_i :

1) $fin_employ(t, a_x, I_i)$,表明 a_x 是 N_i 构建过程的最后一个雇佣节点,记录发送者是 I_i ;

2) $fin_referrer(t, a_i, I_i)$,表明 N_i 在构建过程中提供最后雇佣节点的推荐人是 a_i ,记录发送者是 I_i 。

4.1.2 检查记录

供应网构建算法完成后,供应网 N_i 的请求者 I_i 负责核对 SN 记录库(包括 I_i 曾向 SN 公开的记录,以及 SN 中的其他节点曾公开给 I_i 的记录)。若发现有节点 a_i 同时满足以下情形,则标记 a_i 一次不诚信行为:

- (1)有记录 $relation(\cdot, a_j -> a_i, \cdot)$
- (2)有记录 $visit(t', a_i, I_i')$
- (3)无记录 $fin_referrer(t, a_i, I_i')$
- (4)无记录 $relation(t', a_i -> a_j, I_i')$

条件(1)和条件(4)表明, a_i 与 a_x 间的邻居关系不对称, a_j 曾推荐过邻居 a_i ,但 a_i 没有在某供应网 N_i' 构建过程中推荐 a_j ;设置条件(2)是因为算法 1 的 terminate 前未要求推荐邻居的节点 a_i ,并无机会推荐自己的邻居 a_j ;设置条件 3 是因为对于算法 1 最后雇佣节点的推荐人 a_i ,很可能在 a_i 提供完自己所有邻居之前, N_i' 便结束了构建,也没有机会推荐 a_j 。

检查到 a_i 的一次恶意隐瞒行为之后,公开向 SN 声明 $DisHonest(a_i) += 1$,以增加一条 a_i 非诚信行为记录。同时向 SN 记录库添加记录 $relation(t', a_i -> a_j, I_i')$,以防止该非诚信行为被其他请求者重复标记。

4.2 立约者扣押行为检查机制

SN 中 I_i 发布虚假任务或刻意延长 N_i 的构建过程,依靠扣押立约者,向其他请求者索取献金获利。解决此问题的关键在于,保证请求者 I_i 只能处于 3 个阶段:

- (1)正在与某节点谈判或协调;
- (2)构建 N_i 失败,解散 N_i ;
- (3)构建 N_i 成功, N_i 开始供应。

本文同样通过分布式记账的方式记录请求者的状态供 SN 检查。

4.2.1 公开记账

在 I_i 构建 N_i 的过程中,记账规则如下:

(1) I_i 向 SN 公开自己的任务 t 需求;

(2)算法 1 中的步骤 5 中, $decide(N(t), a_x)$, I_i 的谈判对象 a_x 在谈判开始时向整个 SN 公开记录 $DecideStart(t, I_i, a_x)$,表明 I_i 为完成任务 t 与 a_x 开始谈判,记录发送者是 a_x ;谈判结束时向整个 SN 公开记录 $DecideFinish(t, I_i, a_x)$,表明 I_i 与 a_x 的谈判结束,记录发送者是 a_x 。

(3)算法 2 中的步骤 2 中, $Cooperate(I_i, I_{i2}, a_i, g_j, Q_{temp}, P_{temp})$, I_i 的协调对象 I_{i2} 在协调开始时,向整个 SN 公开记录 $CoopStart(t, I_i, I_{i2})$,表明 I_i 为任务 t 与 I_{i2} 开始协调,记录发

送者是 I_{t_2} ; 协调结束时, 向整个 SN 公开记录 $Coopfinish(t, I_t, I_{t_2})$, 表明 I_t 与 I_{t_2} 协调结束, 记录发送者是 I_{t_2} 。

(4) I_t 与节点 a_x 达成订单时, a_x 向 SN 公开记录 $Order(t, I_t, g_j, q(a_x, g_j, t), a_x)$, 表明 I_t 为了完成任务 t , 与 a_x 达成了关于 g_j 的量为 $q(a_x, g_j, t)$ 的订单, 记录发送者是 a_x ; 相应地, a_x 通过记录 $Cancel(t, I_t, g_j, q(a_x, g_j, t), a_x)$ 声明取消该订单, 记录发送者是 a_x 。

(5) N_t 构建成功后, I_t 马上向 SN 声明 $Succ(t, N_t, I_t)$, 表明成功构建了覆盖任务 t 全部需求的供应网 N_t ; 相应地, 若构建失败, 则 I_t 声明 $Fail(t, N_t, I_t)$, 表明本次任务失败。

4.2.2 检查记录

记录 I_t 状态的信息受 SN 中其他成员的监督, 一旦检查到 I_t 符合以下情形之一, 就标记 I_t 一次非诚信行为:

(1) 某时刻, I_t 谈判与协商信息全部成对出现 (即 $DecideStart(t, I_t, a_x)$ 与 $DecideFinish(t, I_t, a_x)$, $CoopStart(t, I_t, I_{t_2})$ 与 $Coopfinish(t, I_t, I_{t_2})$), 且一定时间内没有更新状态 (即出现新的 $DecideStart(t, I_t, \cdot)$ 或 $DecideFinish(t, I_t, \cdot)$ 记录, 或 I_t 宣布 N_t 构建成功/失败);

(2) N_t 中立约者收到 I_t 向 SN 发出的 $Succ(t, N_t, I_t)$ 或 $Fail(t, N_t, I_t)$ 声明后, 没有被要求开始供应工作或被解散;

(3) 某时刻, 任务 t 的全部需求已由未被 cancel 的订单 $Order(t, I_t, \cdot, q(\cdot, \cdot, t), \cdot)$ 所满足, 却没有声明 $Succ(t, N_t, I_t)$;

(4) I_t 已声明 $Succ(t, N_t, I_t)$, 但已公开的 t 的需求却没有被未 cancel 的订单 $Order(t, I_t, \cdot, q(\cdot, \cdot, t), \cdot)$ 满足。

检查到 I_t 的一次非诚信行为之后, 公开向 SN 声明 $DisHonest(I_t) += 1$, 以增加一条 I_t 非诚信行为记录。单个任务 t 中, I_t 最多被标记一次非诚信行为。

4.3 非诚信行为惩罚机制

对非诚信行为者的惩罚通过信誉机制来完成。SN 中节点 a 的非诚信行为标记数 $DisHonest(a)$ 是衡量 a 信誉的唯一标准, 通过不信任系数 $Distrust(a)$, 来影响节点与其他节点合作时的额外信任代价。

不信任系数的计算公式为 $Distrust(a) = \Psi * DisHonest(a)$, 该系数影响供需订单额外的信任成本。 a 的不信任系数与 a 被标记的非诚信行为次数 $DisHonest(a)$ 成正比, 其中 Ψ 用来描述不信任系数随标记数上升的幅度。

Ψ 的大小需依据 SN 具体结构确定。对于成员较少、技能较少、社交关系不紧密的 SN, 若过度上调不诚信者的不信任系数, 很可能造成该 SN 下的任务 t 都难以找到符合成本要求的供应者, 无法完整地构建 N_t ; 但当 SN 内成员较多、技能丰富、社会关系紧密时, 大幅度上调不诚信者的不信任系数可以更有效杜绝非诚信行为, 而几乎不影响构建供应网的成功率。因此, Ψ 值需要针对不同的 SN 做适当调研后再确定。

5 机制有效性分析

我们将通过分析来证明, 在我们的机制下众包参与者的最优策略是保证诚实。

5.1 恶意隐瞒行为的检查与惩罚机制

对于产品的供应者 a_i 来说收益即为订单量, 其策略目标

是在 SN 的长期活动中获得更多的订单量。为使惩罚效果明显, 我们假设信任系数随标记数上升的幅度 Ψ 很大, 以至于一旦 a_i 被标记不诚信, 将失去未来 SN 中的所有订单。

在我们的机制下, 当某产品 g_j 的供应商 a_i 被请求者 I_t 加入供应网 N_t , 并被要求推荐其邻居, 而邻居 a_j 的 g_j 成本更低时, a_i 面临两种选择:

(1) 选择不诚信, 隐瞒邻居 a_j , 其理由为: 可以防止本次订单被 a_j 替换。

(2) 选择诚信, 向 I_t 推荐低成本邻居 a_j , 其理由如下:

1) 即使推荐了邻居, a_i 的 g_j 也不一定被 a_j 替换 (a_j 此时正被其他请求者雇佣的情况);

2) 即使 a_i 订单被邻居 a_j 替换, 也未必失去全部订单 (a_j 的剩余 g_j 产能 $< a_i$ 当前的订单量 + I_t 当前的剩余未满足 g_j 需求, 即 $\max(a_j, g_j) < q(a_i, g_j, t) + q_{ws}(t, g_j)$ 的情况);

3) 即使不诚信地隐瞒了邻居 a_j , 也不能保证本次订单不会被本次 N_t 构建过程中访问到的其他供应者替换;

4) 根据本文的检查机制, 只要邻居 a_j 被某次 N_t' 构建过程访问并被要求推荐邻居, a_i 的非诚信行为就一定会被发现, a_i 将失去此后 SN 中的全部订单。

因此, 在这种机制下, 尽管 a_i 不诚信可以避免订单被其邻居 a_j 替代, 但考虑到选择 (2) 中理由 3) 的情形, 却不能保证一定获得本订单; 在长期来看, 保持诚信可以在选择 (2) 中理由 1) 与理由 2) 等的情形下在 SN 中持续获得少量订单。长期来看, 保持诚信对高成本供应者来说是占优策略。

此外记账与检查过程均由任务请求者完成, 如实记录与检查有利于维护 SN 中请求者与诚实供应者的利益, 因此任务请求者有充足的动机执行该恶意隐瞒行为检查机制。因此, 本文的机制既是有效的, 也是可行的。

5.2 立约者扣押行为的检查与惩罚机制

对于任务请求者 I_t , 在 SN 中的长期目标为成功组建更多供应网以获得更大的任务收益。同样, 为使得本文惩罚机制的效果明显, 我们假设信任系数随标记数上升的幅度 Ψ 很大, 以至于一旦 I_t 被标记不诚信, I_t 与供应者达成订单时信任成本的提升将导致其无法在 SN 中成功构建任何供应网。

根据本文检查机制, 一旦 I_t 有扣押立约者的嫌疑, 就将立刻被立约者扣押行为检查机制中记录检查的规则发现:

(1) 若 I_t 发布虚假任务, 或为扣押立约者故意不进行谈判, 则该类行为可由检查规则中的第一种情形检查到;

(2) 若 N_t 构建失败或成功后, I_t 仍扣押立约者, 则该类行为可由第二种情形检查到;

(3) 若 N_t 的需求被满足后, I_t 为延长立约者扣押时间, 仍不声明构建成功, 而继续与节点谈判, 则该类行为可由第三种情形检查到;

(4) 若 I_t 为了能够在 N_t 构建成功后继续与节点谈判以延长立约者的扣押时间, 提前虚报任务 t 需求, 该类行为可被第四种情形检查到 (这是因为 I_t 延长扣押时间后, 也须声明 $Succ(t, N_t, I_t)$, 才能将早已构建成功的 N_t 投入供应工作, 此时便可发现订单详情与任务 t 的需求不符)。

由于检查规则, 请求者 I_t 既不能通过立约者扣押行为获利 (其非诚信行为由 SN 实时监督, 未等通过协调索取献金获

利, 便将被发现), 又将因惩罚机制失去此后在 SN 中成功组建供应网的可能。因此, 保持诚信是请求者的占优策略。

此外, 检查规则中的第(2)–(4)条使 I_i 无法公开虚假的 t 需求和虚假的 $Succ(t, N_i, I_i), Fail(t, N_i, I_i)$ 记录, 而其他的记录与检查过程由 SN 中的其他节点完成。如实记录与检查有利于维护其他请求者与供应者的利益, 因此其他节点有充足的动机执行该立约者扣押行为检查机制。因此, 本文的机制既是有效的, 也是可行的。

结束语 本文首先意识到了众包模式供需匹配方法中可能存在的两类非诚信行为, 并针对此问题设计了一套非诚信行为检查与惩罚机制。该机制通过分布式记账的方式保障信息的真实性, 通过核对公开信息来维护社交网络和持续检查个体的不诚信行为, 并通过标记的方式调节非诚信者的信任代价系数, 使其在之后的供需合作中提升代价。在本文的机制下, 个体参与众包的最优策略便是保证诚实。

本文的研究只设计了惩罚机制, 即信任系数的上调规则, 可以进一步考虑对非诚信者的信誉恢复机制, 允许根据非诚信者近期的合作表现逐步恢复信誉。此外, 在本文惩罚机制的设计中, 没有给出不信任系数上调幅度 Ψ 的计算细节, 因为这将涉及对整个 SN 组成结构的评估, 还需要结合统计学方式对真实数据采样并回归分析, 展开范围较大, 也不属于本文研究的内容, 将来可针对该问题展开进一步研究。

参 考 文 献

[1] YUEN M C, KING I, LEUNG K S. A Survey of Crowdsourcing Systems[C]//Proceedings of IEEE International Conference on Privacy, Security, Risk and Trust and IEEE International Conference on Social Computing. Los Alamitos: IEEE Computer Society, 2011: 766-773.

[2] BUCCAFURRI F, LAX G, NICOLAZZO S, et al. Driving Global Team Formation in Social Networks to Obtain Diversity[C]//Proceedings of International Conference of Web Engineering. Berlin: Springer, 2014: 410-419.

[3] CHAMBERLAIN J. Groupsourcing: Distributed Problem Solving Using Social Networks[C]//Proceedings of AAAI Conference on Human Computation and Crowdsourcing. Menlo Park: AAAI Press, 2014: 22-29.

[4] JIANG J, AN B, JIANG Y, et al. Group-Oriented Task Allocation for Crowdsourcing in Social Networks[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2019(99): 1-16.

[5] KARGAR M, AN A, ZIHAYAT M. Efficient bi-objective team formation in social networks[C]//Proceedings of European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases. Berlin Heidelberg: Springer-Verlag, 2012: 483-498.

[6] LYKOURENTZOU I, WANG S, KRAUT R E, et al. Team Dating: A Self-Organized Team Formation Strategy for Collaborative Crowdsourcing[C]//Proceedings of ACM CHI 2016: the top conference for Human-Computer Interaction. New York: ACM Press, 2016: 1243-1249.

[7] ROKICKI M, ZERR S, SIERSDORFER S. Groupsourcing: Team competition designs for crowdsourcing[C]//Proceedings of the

24th International Conference on World Wide Web. New York: ACM Press, 2015: 906-915.

[8] SUN Y, TAN W, HUANG L. A team discovery model for crowdsourcing tasks to social networks[J]. International Journal of Engineering and Technology, 2017, 12(1): 21-44.

[9] HUANG J B, SUN X J, ZHOU Y, et al. Research Survey on Team Formation in Social Networks[J]. Journal of Software, 2017, 28(4): 993-1009.

[10] JIANG J, AN B, JIANG Y, et al. Understanding crowdsourcing systems from a multiagent perspective and approach[J]. ACM Transactions on Autonomous and Adaptive Systems, 2018, 13(2): 1-32.

[11] LIU W W, DENG Z H, CAO L B, et al. Mining Top K Spread Sources for a Specific Topic and a Given Node[J]. IEEE Transactions on Cybernetics, 2015, 45(11): 2472-2483.

[12] ZHANG S J, LU X D, WANG S P, et al. Supply-Demand Matching in Non-Cooperative Social Networks[J]. IEEE Access, 2020, 8: 162458-162466.

[13] LIU Q, LUO T, TANG R, et al. An Efficient and Truthful Pricing Mechanism for Team Formation in Crowdsourcing Markets[C]//Proceedings of IEEE International Conference on Communications. Piscataway: IEEE Press, 2015: 567-572.

[14] WANG W Y, HE Z P, SHI P, et al. Truthful team formation for crowdsourcing in social networks[C]//Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems. Berlin: Springer, 2016: 1327-1328.

[15] JIANG Y C, JIANG J C. Understanding Social Networks From a Multiagent Perspective[J]. IEEE Transactions on Multimedia, 2014, 25(10): 2743-2759.

[16] TRAN-THANH L, STEIN S, ROGERS A, et al. Efficient crowdsourcing of unknown experts using multi-armed bandits[C]//Proceedings of European Conference on Artificial Intelligence. Netherlands: IOS Press, 2012: 768-773.

[17] TRAN-THANH L, HUYNH T D, ROSENFELD A, et al. BudgetFlx: Budget limited crowdsourcing for interdependent task allocation with quality guarantees[C]//Proceedings of International Conference on Autonomous Agents and Multiagent Systems. Berlin: Springer, 2014: 477-484.

[18] WANG W Y, HE Z P, SHI P, et al. Strategic Social Team Crowdsourcing: Forming a Team of Truthful Workers for Crowdsourcing in Social Networks[J]. IEEE Transactions on Mobile Computing, 2019, 18(6): 1419-1432.



ZHANG Shao-jie, born in 1998, master, is a member of China Computer Federation. His main research interests include multiagent systems, machine learning, and reinforcement learning.



LU Xu-dong, born in 1971, Ph.D, lecturer, is a member of China Computer Federation. His main research interests include crowd science, big data technology and intelligent data analysis, medical treatment and health.