Allocating Tasks to Workers with Matching Constraints: Truthful Mechanisms for Crowdsourcing Markets

Gagan Goel
Google, NY
gagangoel@google.com

Afshin Nikzad
Stanford
nikzad@stanford.edu

Adish Singla
ETH Zurich
adish.singla@inf.ethz.ch

ABSTRACT

Designing optimal pricing policies and mechanisms for allocating tasks to workers is central to the online crowdsourcing markets. In this paper, we consider the following realistic setting of online crowdsourcing markets – there is a requester with a limited budget and a heterogeneous set of tasks each requiring certain skills; there is a pool of workers and each worker has certain expertise and interests which define the set of tasks she can and is willing to do. Under the matching constraints given by this bipartite graph between workers and tasks, we design our incentive-compatible mechanism TM-UNIFORM which allocates the tasks to the workers, while ensuring budget feasibility and achieves near-optimal utility for the requester. Apart from strong theoretical guarantees, we carry out experiments on a realistic case study of Wikipedia translation project on Mechanical Turk. We note that this is the first paper to address this setting from a mechanism design perspective.

Categories and Subject Descriptors: H.0 [Information Systems]: General
Terms: Algorithms, Economics, Human Factors, Theory
Keywords: Crowdsourcing, mechanism design, matching constraints, incentive compatibility, procurement auctions

1. INTRODUCTION

How does one design market mechanisms for crowdsourcing when the tasks are heterogeneous and workers have different skill sets? The recent adoption of crowdsourcing platforms such as Amazon's Mechanical Turk on Internet has brought increased attention to the scientific questions around the design of such markets. A central theme is that there is a *requester* who posts tasks to be carried out by a pool of online *workers*. The requester's goal is to maximize the utility derived from the task within her limited budget, while the workers try to maximize their own individual utility by deciding which tasks to perform and at what price.

To give a concrete application (which also becomes a subject of our experimental validation), consider a requester who wants to translate Wikipedia articles into different languages. Here, a tuple of an article's topic and a target language represents a unique task. There is a pool of workers, and based on the language skills and topic expertise, each worker can only translate some articles into some languages, and not all. Mathematically speaking, this results in a bipartite graph between workers and tasks. Also assume that each worker can only do limited number of tasks and has a private minimum cost that she wants to get paid for doing a task. We seek to optimize the requester's utility using incentive-compatible mechanisms that are budget feasible.

We note that we study this problem for the first time from a mechanism design perspective. Previous results either ignored the strategic nature of the workers [4, 1] or they consider simpler models where the notion of the tasks is not present and the utility is tied directly to the workers [5, 6].

WWW'14 Companion,, April 7–11, 2014, Seoul, Korea.
ACM 978-1-4503-2745-9/14/04.
http://dx.doi.org/10.1145/2567948.2577311.

2. THE MODEL

There is a requester with a limited budget B and a set of heterogeneous tasks T. For each task $t \in T$, there is a utility u_t that the requester achieves if that task gets completed. There is a pool of workers P. Each worker $p \in P$ has a private cost c_p which is the minimum payment she is willing to take for doing any task. Each worker has certain skill sets and interests which makes her eligible to do only certain tasks, and not all. We model these constraints with a bipartite graph $G(P, T)$ where an edge $e = (p, t)$ represents the notion that worker p can do task t. We assume a *large market* which represents the notion that the pool of workers is large enough that no single worker can affect the market outcome significantly.

For simplicity, we assume that a worker can do only one task and a task needs to be done only once. However, our mechanisms easily extend for many-to-many assignments as well (where each task needs to be done several times and each worker can do multiple tasks) by simply creating multiple copies of a worker or a task. More interestingly, in the many-to-many setting, we can also handle the case when the utility of doing a task is a non-decreasing concave function of the number of times that the task is done.

The goal is to design a mechanism that solicits bids from workers, and outputs a matching or assignment that represents a set of recruited workers and tasks allocated to them, as well as a payment for each recruited worker. We are interested in mechanisms that satisfy: i) *Truthfulness*, *i.e.* reporting the true cost should be the dominant strategy of the workers, ii) *Budget-feasibility*, *i.e.* the total payments shouldn't exceed the budget B. The mechanism should achieve the above two properties while trying to maximize the total utility obtained from the tasks that get allocated.

3. THE MECHANISM: TM-UNIFORM

The key concept in the mechanism is a *buck per bang* rate r representing the payment that the mechanism is willing to pay per unit of utility *i.e.* if a worker is assigned a task with utility u, then it will be paid $r \cdot u$. The buck per bang rate of an edge $e = (p, t)$, denoted by $\mathsf{bb}(e)$, is defined by $\frac{c_p}{u_t}$. Also, let $G(r)$ be the copy of graph G which only contains edges $E(G)$ with rate at most r.

Mechanism 1 starts with $r = \infty$ and it gradually decreases the rate r. Let $m = |E(G)|$ and $e_1, \ldots, e_m$ be a list in which the edges are sorted *w.r.t.* their buck per bang in decreasing order, *i.e.* for e_i and e_j, we have $i \leq j$ iff $\mathsf{bb}(e_i) \geq \mathsf{bb}(e_j)$. For any fixed r, it constructs the graph $G' = G(r)$ and calls Procedure FindMatching to find a matching or assignment $\mathsf{M} \subseteq E(G')$ in G'. Procedure FindMatching takes as input a fixed permutation σ of the nodes in P. Then, the nodes in P are visited one by one in the order of appearance in σ. When p is visited, the mechanism assigns p to a task t which has the highest utility among all the tasks that can be currently assigned to p. Let M denote the matching returned by the procedure after visiting all the nodes in P. *Cost* of M, denoted by $c(\mathsf{M})$ is defined by $\sum_{(p,t)\in\mathsf{M}} c_p$. Also, *utility* of M, denoted by $u(\mathsf{M})$ is defined by $\sum_{(p,t)\in\mathsf{M}} u_t$. If $r \cdot u(\mathsf{M}) > B$, then mechanism decrease the rate r slightly and repeat this procedure for the new r; otherwise, it stops.

Procedure FindMatching

input : Graph $G(P,T)$, Permutation σ
output: A matching in G
$\mathsf{M} \leftarrow \emptyset$;
for $i \leftarrow 1$ **to** $|P|$ **do**
  Find the task t with the highest utility which is available for $\sigma(i)$;
  $\mathsf{M} \leftarrow \mathsf{M} \cup (\sigma(i), t)$;
end
Return the matching M;

Mechanism 1: TM-UNIFORM

input : Graph $G(P,T)$, Budget B, Permutation σ
output: A matching in G
$G' = G$;
for $i \leftarrow 1$ **to** m **do**
  $\mathsf{M} = \text{FindMatching}(G', \sigma)$;
  if $\mathsf{bb}(e_i) \cdot u(\mathsf{M}) \leq B$ **then**
    $r \leftarrow \min\left(\frac{B}{u(\mathsf{M})}, \mathsf{bb}(e_{i-1})\right)$;
    break;
  end
  $E(G') \leftarrow E(G') - \{e_i\}$;
end
Return M as the final matching;
Make the uniform payments with rate r.

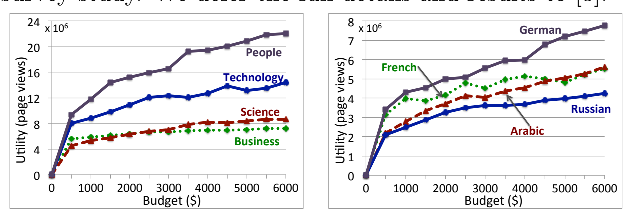
The mechanism uses a uniform payment scheme, *i.e.* paying each worker $r \cdot u_{\mathsf{M}(p)}$ (where $\mathsf{M}(p)$ denotes the task assigned to p, possibly equal to $\emptyset$). With this payment, mechanism TM-UNIFORM satisfies truthfulness in a weaker form, which we call *one-way-truthfulness* (*i.e.* players only have incentive to report costs lower than their true cost). This uniform payment scheme makes it easy to analyze the performance of the mechanism. We can modify TM-UNIFORM and make it fully truthful by slightly modifying the payment rule. In the new payment rule, each recruited worker (*winner*) is paid the threshold payment, *i.e.* the highest cost that it could report and still remain a winner. Now, we state our main results (complete proofs are given in [3]):

Theorem 1. TM-UNIFORM *is budget feasible, individually rational, truthful, and is* 3*-approximate compared to the optimum solution (which assumes access to the true costs).*

Theorem 2. *Using randomization (inspired from [2]), we can improve the approximation ratio to* $\frac{2e-1}{e-1}$.

4. EXPERIMENTAL EVALUATION

Our experiments are based on a project that seek to translate 5,000 most popular weekly pages on English Wikipedia to the 10 most widely used languages on the internet. Each page is associated with one of the 25 different topics based on the top level classification topics from Wikipedia. Thus, we have a total of 250 unique tasks (*number of topics times the number of target languages*), with a total of 50,000 HITs (human-intelligent tasks) to be performed (*number of source pages times number of target languages*). The utility of a HIT is proportional to the *view count of source page times the population of target language.* Using survey study, we elicited following preferences from 1000 workers on Mechanical Turk: i) page topics and languages of interest (which together define worker's skills), ii) bid per HIT and iii) number of HITs to perform. Figure 1 illustrates the utility obtained by running our mechanism offline based on the Wikipedia data and workers' perferences collected from the survey study. We defer the full details and results to [3].



(a) Utility per topic (b) Utility per language

Figure 1: Results illustrate market dynamics by showing the utility acquired per different topic and language as budget is varied. In **(b)**, *French* language acquires higher utility in the beginning, attributed to bigger pool of available workers (65.7% for *French* vs 27% *Arabic* on MTurk). Eventually *Arabic* language catches up because of higher utilities associated with HITs attributed to larger user base of the language (59.8 million for *French* vs 65.4 million for *Arabic*)

5. DISCUSSION AND FUTURE WORK

We would like to point out that our mechanism takes as input a permutation on the workers that can serve as a useful tool to manipulate the outcome (such as using worker's ratings). As future work, we will look at the generalizations where workers can have varying costs for different tasks and when tasks require multiple workers for completion.

6. REFERENCES

[1] 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 *WWW*, 2013.

[2] G. Goel and A. Mehta. Online budgeted matching in random input models with applications to adwords. In *SODA*, 2008.

[3] G. Goel, A. Nikzad, and A. Singla. Matching workers expertise with tasks: Incentives in heterogeneous crowdsourcing markets. In *NIPS Workshop on Crowdsourcing*, 2013.

[4] C.-J. Ho and J. W. Vaughan. Online task assignment in crowdsourcing markets. In *AAAI*, 2012.

[5] Y. Singer. Budget feasible mechanisms. In *FOCS*, 2010.

[6] A. Singla and A. Krause. Incentives for privacy tradeoff in community sensing. In *HCOMP*, 2013.