Human Computation and Crowdsourcing: Works in Progress and Demonstration Abstracts
An Adjunct to the Proceedings of the Second AAAI Conference on Human Computation and Crowdsourcing

Contextual Procurement in Online Crowdsourcing Markets

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

Ian Lienert
ETH Zurich
ian.lienert@lawecon.gess.ethz.ch

Gábor Bartók
ETH Zurich
bartok@inf.ethz.ch

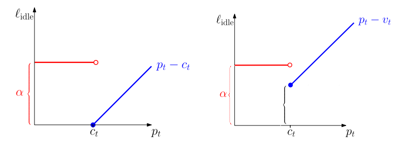
Andreas Krause
ETH Zurich
krausea@ethz.ch

Abstract

Designing pricing mechanisms for recruitment of workers is a central challenge in online crowdsourcing markets. We consider a novel and realistic setting for such markets, where the private cost of the workers and utility derived from them is unknown to the mechanism; however, a set of the workers' features can be observed before making the price offer. How should the offered price be adapted to maximize the utility from recruited workers, while minimizing the cost of payments and the idling cost of failure to recruit? In this paper, we address these questions by formulating the problem as a *contextual partial monitoring game*, a generic framework for online learning problems that allows to deal with complex feedback structure. We present simulation results comparing our approach to the classical contextual bandit approach, demonstrating the complexity of the problem and the need for the partial monitoring framework.

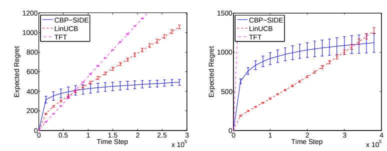
Introduction

The recent adoption of crowdsourcing markets on the Internet (such as Amazon's Mechanical Turk, Clickflower, *etc.*) has created numerous opportunities for outsourcing tasks to online "workers". The principal agent or "requester" who posts the tasks generally has limited budget as well as time constraints and aims to maximize the utility derived from the task. The workers in such markets are diverse and often act strategically in aiming to maximize their profit. Further, the system may have very limited information about them, making it difficult to infer their private cost and potential utility derived from their recruitment. These challenges have brought increased attention to the scientific questions around the design of pricing mechanisms for recruitment of workers in such markets. A series of recent results (Singer 2012; Singla and Krause 2013b; 2013a) have proposed the use of budget feasible procurement auctions to design market mechanisms and pricing policies for crowdsourcing. However, these results are limited and not broadly applicable to more realistic and complex scenarios – the existing mechanisms either assume that the utility is equal across all workers, or that the utility of a worker can be inferred by the mechanism before making the offer. Consider tasks such as translating web documents or image tagging. Here, the utility derived from a worker and the private cost could possibly depend on the demographics. In realistic settings, the mechanism may have access to some of these features that could provide cues for the worker's utility and private cost. Given these contextual features, how can a mechanism learn to adapt the prices to be offered so as to maximize the utility? This is the fundamental question we address in this paper.



(a) $v_t \geq c_t$ (b) $v_t < c_t$

Figure 1: Loss function



(a) Small idling cost $\alpha = 0.15$ (b) Large idling cost $\alpha = 0.75$

Figure 2: Simulation Results

Contextual Partial Monitoring Problem

Model and protocol. The mechanism interacts with the workers sequentially in discrete timesteps denoted by t. Each worker w_t has a private cost $c_t \in \mathcal{C}$ and is associated with a utility $v_t \in \mathcal{V}$ that the mechanism would derive from recruitment. Here $\mathcal{C}$ and $\mathcal{V}$ are the sets of possible prices and utilities, assumed to be in same units. Both c_t and v_t are unknown, however a set of contextual features $x_t \in \mathcal{X}$ (such as demographics of the worker) are observable to the mechanism before making the payment offer. The mechanism computes and offers a price $p_t \in \mathcal{C}$. The worker accepts if $c_t \leq p_t$ and rejects otherwise. Upon acceptance, the worker completes the task, receives a payment of p_t, the requester gains the utility v_t and mechanism receives as feedback the utility value v_t. The actual cost c_t is never revealed.

Loss function and regret minimization. The explicit goal of the requester is modeled as a loss function $\ell(c, v, p)$ that the mechanism aims to minimize. Budget constraints of the requester are modeled as the price inefficiencies of offering higher payment compared to the optimal price that could have been offered at time t. Time constraints are modeled through fixed idling cost α of failure to recruit a worker at timestep t because of offering lower payment than the true cost. The particular loss function we consider in our work is shown in Figure 1(a),1(b) and is given by $\ell_{\text{idle}}(p_t, c_t, v_t) = \max(p_t - v_t, p_t - c_t)\mathbb{I}_{p_t \geq c_t} + \alpha\mathbb{I}_{p_t < c_t}$, where $\mathbb{I}$ is the indicator function. The goal of the mechanism is to be able to learn a latent mapping K from contextual features $\mathcal{X}$ to workers' cost and utilities $\mathcal{C} \times \mathcal{V}$, and perform competitively w.r.t the oracle that offers prices with full knowledge of K. This is captured by the "regret" of the mechanism given by $R_M = L_M - L_O$ where $L_M = \sum_{t=1}^{T} l_t$ is the cumulative loss of the mechanism and L_O denotes the loss of the oracle. The goal is to design a mechanism where the average regret approaches zero asymptotically, i.e., $\lim_{T\to\infty} {}^{R_M}/_T = 0$.

Comparison to multi-armed-bandits (MAB). In our model, the exact loss incurred at each timestep, l_t, cannot be directly computed by the mechanism as it depends on c_t and v_t. This is exactly the reason, why the class of problems considered here is more difficult and generic than what can be modeled by the MAB framework (Li et al. 2010). While simpler crowdsourcing markets have been modeled through the MAB (Singla and Krause 2013b), the complex feedback structure of our model calls for a more generic framework.

Partial monitoring framework. Many realistic settings, as in our model, have complex and limited feedback structure thus concealing the incurred losses. Partial monitoring games (Bartók, Zolghadr, and Szepesvári 2012) provide a powerful model for an online learner acting in such an environment. Bartók and Szepesvári (2012) present an algorithm CBP-SIDE for *contextual* partial monitoring games where the learner is also provided with some side information (context) at each timestep. We present the necessary components to define such a game w.r.t our problem, the complete implementation details are presented by Lienert (2014).

A finite stochastic partial monitoring game with linear side information is a game $\mathbf{G} = (\mathcal{N}, \mathcal{S}, \mathbf{L}, \Sigma, \mathbf{H}, K)$ played between a learner (the mechanism executed on behalf of requester) and an environment (capturing the interaction with the workers) over the course of a finite number of rounds T. The set of actions available to the mechanism, $\mathcal{N} = \{1, \cdots, N\}$, is the set of price offers $\mathcal{C}$ that could be made by the mechanism. $\mathcal{S} = \{1, \cdots, S\}$ denote the possible values of outcome of unknown states of the environment at each timestep and is given by $\mathcal{S} = \mathcal{C} \times \mathcal{V}$ referring to the possible values of cost and utility. $\mathbf{L} \in \mathbb{R}^{N \times S}$ is the *loss matrix* associating each action-outcome pair with a loss in $\mathbb{R}$ and is completely defined by the loss function $\ell_{\text{idle}}(p_t, c_t, v_t)$.

The feedback alphabet Σ defines a set of symbols denoting the possible feedback that could be observed by the mechanism. In our case, the feedback observed is two-fold: *i)* first, the worker's decision about the acceptance (y) or rejection (n) of the offer is observed, *ii)* then, in case of acceptance, the utility v of the worker is observed. Hence, Σ is given by $\{n\} \cup (\{y\} \times \mathcal{V})$. The *feedback matrix* $\mathbf{H} \in \Sigma^{N \times S}$ associates the pair of action (p_t) and outcome (c_t and v_t) with a feedback symbol from the alphabet Σ as discussed above. At each time step t, the mechanism observes side information $x_t \in \mathbb{R}^D$ which completely determines the distribution over outcomes the environment plays according to. We assume $\|x_t\|_1 = 1$. The mapping from side information space to the S-dimensional probability simplex $\Delta_S \subset \mathbb{R}^S$ is realized by the stochastic matrix $K \in \mathbb{R}^{S \times D}$. Note that the mechanism has full knowledge of $\mathcal{N}, \mathcal{S}, \mathbf{L}, \Sigma, \mathbf{H}$ and the goal is to learn K. The oracle also has the full knowledge of K.

Results

We now provide results for simulation experiments. We considered the set of prices as $\mathcal{C} = \{0, 1/2, 1\}$ and three equally-spaced utility values $\mathcal{V} = \{1/3, 2/3, 1\}$. We vary the idling cost from $\alpha = 0.25$ to $\alpha = 0.75$. We considered $D = 9$ for side information (given by $|\mathcal{C}| \times |\mathcal{V}|$), generated uniformly at random from the probability simplex and mapped to an outcome distribution using a predefined linear stochastic matrix K, unknown to the mechanism. We compare the results against TFT and LINUCB as baselines. TFT is naive algorithm we developed that does not learn anything about the outcome distribution, and simply increments or decrements the offered price for a given value of side information, based on the rejection or acceptance of the offer for the same side-information previously. Secondly, we also applied the contextual bandit algorithm LINUCB. As the feedback in our model is less than what is required by the bandit algorithm, applying LINUCB requires approximating bandit feedback from observed variables. We defer the details to Lienert (2014). The results are depicted in Figure 2(a) and 2(b) for $\alpha = 0.25$ and $\alpha = 0.75$. TFT performs badly for both settings and the regret of CBP-SIDE grows sublinearly and faster compared to LINUCB. Also, we note that the case of larger idling cost $\alpha = 0.75$ has higher regret for the algorithms compared to $\alpha = 0.25$. These results demonstrate the complexity and need for partial monitoring framework for such models with limited feedback. Our proposed framework is a step forward in learning and designing mechanisms for more realistic crowdsourcing markets.

Acknowledgments. This research is supported in part by Nano-Tera.ch project "OpenSense2" with Swiss Confederation financing.

References

Bartók, G., and Szepesvári, C. 2012. Partial monitoring with side information. In *Algorithmic Learning Theory*, 305–319.

Bartók, G.; Zolghadr, N.; and Szepesvári, C. 2012. An adaptive algorithm for finite stochastic partial monitoring. In *ICML*.

Li, L.; Chu, W.; Langford, J.; and Schapire, R. E. 2010. A contextual-bandit approach to personalized news article recommendation. In *WWW*.

Lienert, I. 2014. Exploiting side information in partial monitoring games. http://www.inf.ethz.ch/personal/adish.singla/files/lienert.pdf. MSc Thesis.

Singer, Y. 2012. How to win friends and influence people, truthfully: Influence maximization mechanisms for social networks. In *WSDM*.

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

Singla, A., and Krause, A. 2013b. Truthful incentives in crowdsourcing tasks using regret minimization mechanisms. In *WWW*.