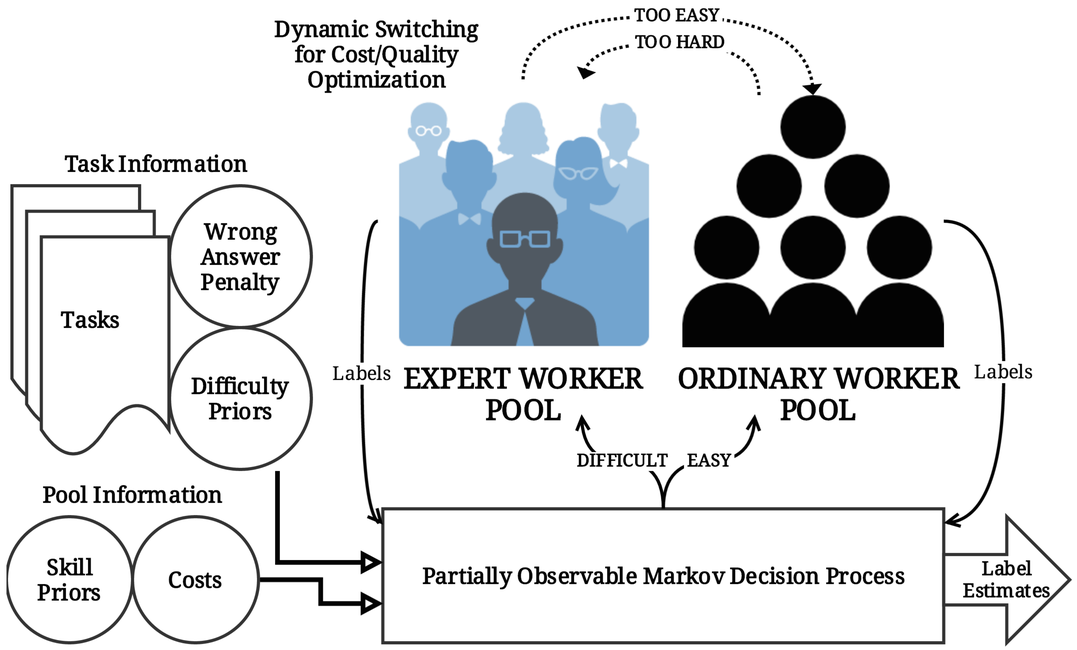
POMDP-Based Worker Pool Selection for Crowdsourcing

Shreya Rajpal, Karan Goel and Mausam
Indian Institute of Technology - Delhi



Problem Overview and Approach

Crowdsourced settings,

- Multiple worker pools with different expertise | MTurk (Master Qualification v/s Normal)
- Different asking prices for more skilled workers | MTurk (higher suggested price for Master)
- Can't always choose best worker for task (pull-style crowdsourcing *i.e.* worker chooses task)
- Push-style crowdsourcing can be more effective (high quality by picking best worker for task)

How can we combine the advantages of push crowdsourcing ideas in pull (MTurk) platforms? Worker Pool Selection! A worker pool is a set of workers with similar quality.

- Use decision theoretic approaches (POMDP) to continuously optimize selection of worker pool
- Choose an expert, higher cost worker pool only when task is harder, otherwise normal pool
- POMDP outperforms baselines that use a single worker pool (improves cost-quality tradeoff)

Key differences of our work from other task-routing papers are

- Different worker pools with heterogenous costs
- Unsupervised tracking of Question difficulty, Individual worker skill, Average worker pool quality
- Dynamic switching, sensitive to Question difficulty | Money already spent | Confidence in answer

Related work does not address this setting

- Donmez, et al. (2009; 2010) (equal difficulty tasks, equal worker costs)
- Karger, et al. (2011; 2013; 2014) (equal difficulty tasks, non-adaptive routing)
- Shahaf & Horvitz (2010); Bragg et al. (2014) (free volunteers)
- Ho et al. (2012; 2013) (equal worker costs, supervised learning of worker skill, workers announce no. of tasks they will do which is non-realistic)

POMDP-Based Framework

Our model extends the work of Dai et al. (2013).

Generative model for worker accuracy

$$a(d,\gamma) = \frac{1}{2}(1 + (1-d)^{\gamma})$$

$d \in [0,1]$: question difficulty, $\gamma \in [0,\infty)$: worker skill parameter

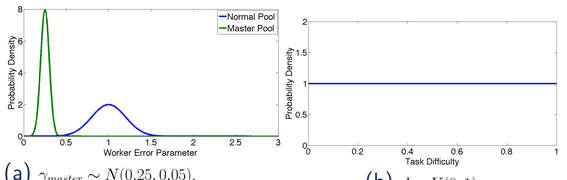
For k worker pools, $\mathbf{wp_1}, \ldots, \mathbf{wp_k}$, our POMDP is,

- $\mathcal{S} = \{(d,v) | d \in [0,1], v \in \{0,1\}\}$ where d is task difficulty and v is true answer.
- $\mathcal{A} = \{$*query* wp_1, *query* wp_2, ..., *query* wp_k, *submit true*, *submit false* $\}$
- $\mathcal{R} : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ contains cost of asking each worker pool, and penalty for submitting incorrect answer.
- $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0,1] = ((d,v), a, (d,v)) \mapsto 1$. All other probabilities are 0.
- $\mathcal{O} = \{true, false\}$ is a 0/1 worker response.
- $\mathcal{P} : \mathcal{S} \times \mathcal{O} \rightarrow [0,1]$ is defined by our generative model.

Unsupervised tracking of individual worker skill and task difficulty is done using an adaptation of Whitehill et al. (2009)'s EM algorithm.

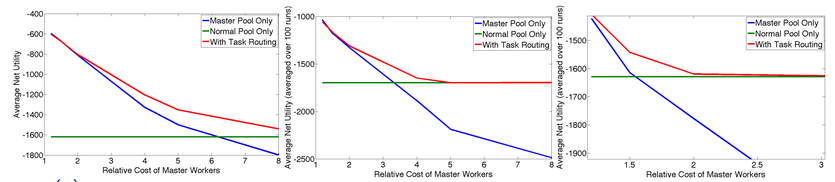
Simulation Experiments on Synthetic Data

1) Simple Setting: Utility Comparison to Single Pool Baselines



(a) $\gamma_{master} \sim N(0.25, 0.05)$, $\gamma_{normal} \sim N(1.0, 0.25)$ (b) $d \sim U(0,1)$

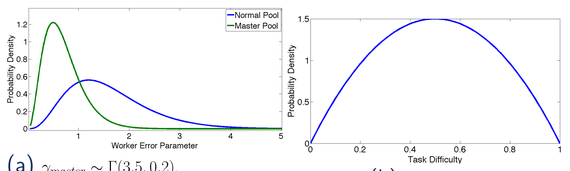
Figure: Distributions used represent a simple setting with reliable experts, and tasks of all difficulties.



(a) $\gamma_{master} \sim N(0.25, 0.05)$ (b) $\gamma_{master} \sim N(0.50, 0.10)$ (c) $\gamma_{master} \sim N(0.75, 0.05)$

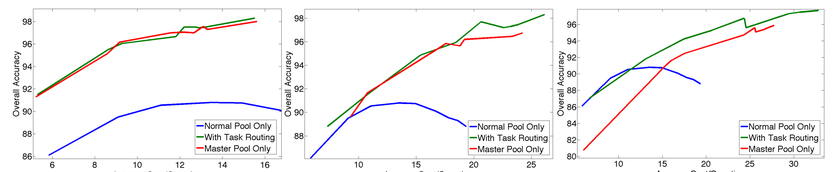
Figure: Task Routing gives better Average Net Utility, regardless of inter-pool skill difference, or relative costs. $\gamma_{normal} \sim N(1.0, 0.25)$.

2) Realistic Setting: Cost-Quality Comparison to Single Pool Baselines



(a) $\gamma_{master} \sim \Gamma(3.5, 0.2)$, $\gamma_{normal} \sim \Gamma(4.0, 0.4)$ (b) $d \sim \beta(2,2)$

Figure: Distributions used represent a realistic setting with less separated pools, and tasks of medium difficulty.



(a) $Cost_{master} = 2$ (b) $Cost_{master} = 4$ (c) $Cost_{master} = 6$

Figure: Task Routing gives better accuracy or lower cost, regardless of inter-pool pricing. $\gamma_{master} \sim \Gamma(3.5, 0.2)$, $\gamma_{normal} \sim \Gamma(4.0, 0.4)$.

Synthetic data demonstrates

- Cost-sensitivity
  - Switches to normal when master pool too expensive
  - Switches to master when master pool is cheap
  - Maximum improvement when intermediate pricing
- Skill-sensitivity
  - Switches to normal more easily when inter-pool skill differential is low
- 20% less for 95% accuracy
  - Boosts normal pool performance by asking experts
  - Doesn't overpay & ask experts for easy tasks

Live Experiments (MTurk)

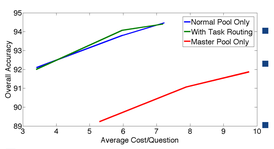


Figure: Master pool performs worse than normal, so all tasks are routed to the normal pool.

- Entity Disambiguation Task
- 150 Questions | Varied difficulties
- **Master Pool** Categorization Masters | > 5000 HITs | > 98% approval rate | 2.4¢/question
- **Normal Pool** Non-Masters | > 100 HITs | > 95% approval rate | 1.6¢/question
- Master pool is *worse* than normal pool
- POMDP does *not* route to master pool, as expected
- Questions quality of MTurk master workers despite their higher price