Incentivizing the Biased Requesters: Truthful Task Assignment Mechanisms in Crowdsourcing

Jia Xu, Hui Li, Yanxu Li, Dejun Yang, Tao Li

School of Computer, Jiangsu Key Laboratory of Big Data Security & Intelligent Processing, Nanjing University of Posts & Telecommunications
Crowdsourcing with Biased Requesters

I intend to hire native Hungarian speakers to order Hungarian to English translations.

I wish to allocate the research projects to the students who are interesting in.

I expect to assign the mobile crowdsourcing tasks to the workers who are close by the specific locations.

Preference over workers!
Crowdsourcing Process

Designing truthful task assignment mechanisms to maximize the total value.

Each consists of a task and a preference set.

Preference set: a set of compatible workers.
Challenges

- Compatibility
- Strategic action
- Workload feasibility
Contributions

First work to design truthful assignment mechanisms for the crowdsourcing systems with biased requesters.

Formulate the Valuation Maximizing Assignment (VMA) problem in three different models.

Design an assignment mechanism for each of these models to solve the VMA problem. We show that the designed mechanisms satisfy four desirable properties: computational efficiency, workload feasibility, preference (universal) truthfulness, and constant approximation.
II-Model: Identical workload Identical value

**Objective Function:**

\[
\max_A v(A) = \sum_{k \in W} |A^k| = \sum_{i \in R} |A_i|
\]

**Constraints:**

1. \(|A^k| \leq 1, \forall k \in W\)
2. \(|A_i| \leq 1, \forall i \in R\)
3. \(A \in \{(i,k) | i \in R, k \in P_i\}\)
It Looks So Easy?!

Requesters  Tasks  Workers

... ... Maximum Bipartite Matching
However

The VMA Problem

Maximum Bipartite Matching

untruthful

Ford Fulkerson Algorithm
TAM-II

Sort the task-worker pairs

Compute the size of the maximum matching

Remove the allocations which cannot reduce the size of maximum matching

Input: Worker Set $W$, Request Set $B$

1: $A \leftarrow \emptyset$

2: Represent all pairs $(i, k), i \in R, k \in P_i$ as $(1,1),(1,2),...,(2,1),(2,2),..., (n, m)$, and the sequence is denoted by $\mathcal{H}$;

3: $\mathcal{H}' \leftarrow \mathcal{H}$;

4: $N \leftarrow MBM(\mathcal{H})$;

5: for all $j \in \mathcal{H}$ in order do

6: $N' \leftarrow MBM(\mathcal{H}' \setminus \{j\})$;

7: if $N' \geq N$ then

8: Remove $j$ from $\mathcal{H}'$;

9: end if

10: end for

11: $A \leftarrow \mathcal{H}'$;

12: return(A)
Generalize to the Non-identical Value Case

**Objective Function:**

\[
\max_A v(A) = \sum_{(i,k) \in A} v_i^k = \sum_{(i,k) \in A} F(a_i) * I_k
\]

**Constraints:**

(1) \[|A^k| \leq 1, \forall k \in W\]

(2) \[|A_i| \leq 1, \forall i \in R\]

(3) \[A \in \{(i,k) \mid i \in R, k \in P_i\}\]
How About Hungarian Method?

The VMA Problem in IN-Model

Maximum Weighted Bipartite Matching Problem

Hungarian Method
Check the Truthfulness of Hungarian Method

An Example

Hungarian method is untruthful

Requester2 lies
A Greedy Algorithm—TAM-IN

Input: Worker Set $W$, Request Set $B$, Effort Indicators $I$

1: Sort all pairs $(i, k)$, $i \in R, k \in P_i$ based on $v_i^k$ in nonincreasing order and the sequence is denoted by $J$;

2: $A \leftarrow \emptyset$;
3: for all $j \in J$ in order do
4:   if $A \cup \{j\}$ is a matching on $G(R, W, J)$ then
5:     $A \leftarrow A \cup \{j\}$;
6:   end if
7: end for
8: return $(A)$;
Non-identical Workload Non-identical Value Case

Objective Function:

$$max_A v(A) = \sum_{(i,k) \in A} v_i^k$$

Constraints:

(1) $$\sum_{i \in A^k} c_i \leq C_k, \forall k \in W$$

(2) $$|A_i| \leq 1, \forall i \in R$$

(3) $$A \in \{(i,k) | i \in R, k \in P_i\}$$
Try greedy Assignment Mechanisms

**GREEDY-VALUE**
Select the task-worker pairs iteratively in nonincreasing order of value

**GREEDY-DENSITY**
Select the task-worker pairs iteratively in nonincreasing order of the ratio of the value to the workload

The VMA problem in the NN-Model is NP-hard since it contains a MKP (Multiple Knapsack Problem)
How Good are Greedy Algorithms?

- Computational efficiency
- Workload feasibility
- Preference truthfulness
- Approximation
Let $\varepsilon > 0$, the approximation ratio of GREEDY-VALUE tends to infinite.

Assume that $\alpha > 1, \alpha - \varepsilon < 1, \varepsilon \in (0, 1/2]$.

Since $1 > \alpha - \varepsilon$, $v(A_{\text{value}}) = 1$

$v(A_{\text{opt}}) = \left\lfloor \frac{1}{\varepsilon} \right\rfloor (\alpha - \varepsilon)$.

Let $\varepsilon$ be sufficiently close to 0, the approximation ratio of GREEDY-VALUE tends to infinite.
Approximation Ratio of GREEDY-DENSITY

A Example

Assume $\alpha > 1, \alpha \varepsilon < 1, \varepsilon \in (0,1)$.

Since $\frac{\alpha \varepsilon}{\varepsilon} = \alpha > 1$, $v(A_{\text{density}}) = \alpha \varepsilon$

Since $\alpha \varepsilon < 1$, $v(A_{\text{opt}}) = 1$

If we let $\varepsilon$ be sufficiently close to 0, the approximation ratio of GREEDY-DENSITY tends to infinite.

Bad News

There is no upper bound of the approximation ratio for either GREEDY-VALUE or GREEDY-DENSITY.
A Random Method—TAM-NN

Input: Worker Set $W$, Request Set $B$, Effort Indicators $I$, Workload Constraints $C$

1: Generate a random number $o$ from the uniform distribution on the interval $[0,1]$;
2: $A \leftarrow \emptyset$;
3: if $o \leq 1/2$ then
4: $A \leftarrow$ GREEDY−VALUE($W, B, I, C$);
5: else
6: $A \leftarrow$ GREEDY−DENSITY($W, B, I, C$);
7: end if
8: return($A$)
Theorem 1. TAM-II is computationally efficient, workload feasible, preference truthful and optimal for VMA problem in the II-Model.

Theorem 2. TAM-IN is computationally efficient, workload feasible, preference truthful and 2-approximate in the IN-Model.

Theorem 3. TAM-NN is computationally efficient, workload feasible, preference truthful and 4-approximate in the NN-Model.
The average approximation ratio of TAM-IN is 1.27

On average, TAM-NN can output 8.45% and 12.38% more value than GREEDY-VALUE and GREEDY-DENSITY, respectively.
B. Running Time

TAM-IN only takes averagely 9.52% of time required by HANGARIAN in all cases
We have investigated the task assignment incentive mechanisms for the crowdsourcing system with biased requesters.

We have studied three models of crowdsourcing and formulated the VMA problem for each model. We presented the task assignment mechanisms for all three models, and proved that they are computationally efficient, workload feasible, preference (universally) truthful and constant approximate.

Extensive results are presented to verify our theoretical analysis.