Do Extra Dollars Paid-Off? - An Exploratory Study on TopCoder

Lili Wang
Department of Computer Science and Technology
Ocean University of China
Qingdao 266100, China
wanglili@stu.ouc.edu.cn

Yong Wang
Department of Computer Science and Technology
Ocean University of China
Qingdao 266100, China
wangyong@ouc.edu.cn

Abstract

In general crowdsourcing, different task requesters employ different pricing strategies to balance task cost and expected worker performance. While most existing studies show that increasing incentives tend to benefit crowdsourcing outcomes, i.e. broader participation and higher worker performance, some reported inconsistent observations. In addition, there is the lack of investigation in the domain of software crowdsourcing. To that end, this study examines the extent to which task pricing strategies are employed in software crowdsourcing. More specifically, it aims at investigating the impact of pricing strategies on worker's behaviors and performance. It reports a conceptual model between pricing strategies and potential influences on worker behaviors, an algorithm for measuring the effect of pricing strategies, and an empirical evaluation on 434 crowdsourcing tasks extracted from TopCoder. The results show that: 1) Strategic task pricing patterns, i.e. under-pricing and over-pricing are prevalent in software crowdsourcing practices; 2) Overpriced tasks are more likely to attract more workers to register and submit, and have higher task completion velocity; 3) Underpriced tasks tend to associate with less registrants and submissions, and lower task completion velocity. These observations imply that task requesters can typically get their extra dollars investment paid-off if employing proactive task pricing strategy. However, it is also observed that it appears to be a counter-intuitive effect on the score of final deliverable. We believe the preliminary findings are helpful for task requesters in better pricing decision and hope to stimulate further discussions and research in pricing strategies of software crowdsourcing.

CCS Concepts •**Software engineering** → Software management → Software development process management

Keywords crowdsourcing, task award, pricing strategy, worker behaviors, worker performance

1 Introduction

The price you set for a product or service has a very significant effect on how the consumer behaves. Intensive studies on motivation patterns of crowdsourcing workers have reported that monetary prize set and funded by task requesters, is one of the top motivating factors to attract potential workers to participate task competition [1]. Therefore, reasonable task award is the basis to attract qualified participants and consequently, satisfactory submissions [2]. Moreover, different task requesters may employ different pricing strategies to balance task cost and degree of competitions w.r.t. particular circumstances. For example, in order to encourage a higher number of solvers and higher ability level of winners, it is better to set higher rewards, as Shao et al. stated in [3]. However, requesters will minimize payment of tasks when facing budget constraints [4].

Several existing work reported various models to address software crowdsourcing task pricing issues, e.g. Mao et al. [5] and Turki et al. [6]. They are useful in helping task requesters to estimate the reasonable or nominal task price following traditional software cost modeling approach. However, in software crowdsourcing, task pricing is far more than cost estimation. Fig. 1 shows a typical pricing decision process on the purchase of consumer goods, including seven steps from strategy determination to adapt appropriate price structure to balance demand and cost [7]. It is a logical approach employed by consumer-goods firms for setting profitable price. We hypothesize that software crowdsourcing tasks, i.e. the purchase of technical services, may follow a similar, though frequently in unconscious manners. More specifically, task requesters may choose to add additional incentives if they perceive low worker supply or high competition from other requesters. In such cases, they not only need to estimate a price (step 5), but also to need to adopt appropriate pricing strategy to ensure the pricing structure are competitive enough to meet the variations in worker supply by setting a price level (step 6).

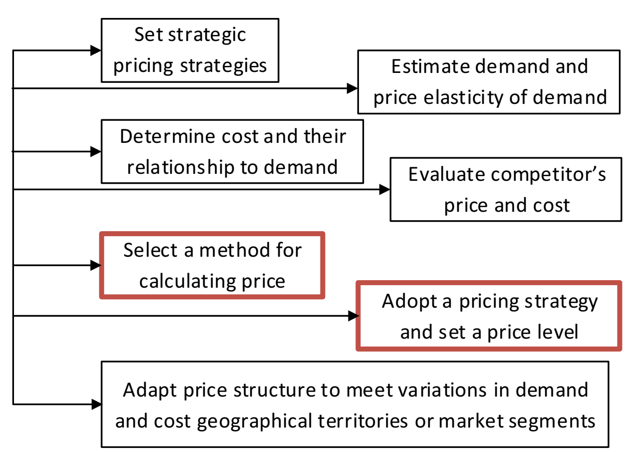

Figure 1. Price setting decision process

CSI-SE'18, May 27, 2018, Gothenburg, Sweden

ACM ISBN 978-1-4503-5733-3/18/05$15.00
https://doi.org/10.1145/3195863.3196958

In practice, task requesters probably modify the predictive price out of different considerations, e.g., they may reduce task award slightly to save cost, or properly raise the price to achieve broader competition. So it is highly necessary for requesters to understand "How to effectively set the strategic award of crowdsourcing task?", i.e. which pricing strategy should be employed to achieve the anticipated worker performance. In this context, it is more critical to answer questions like "what is the right price?" than "what is the nominal price?".

In terms of the relationship between task price and worker performance, there is a common view that higher payment and rewards encourage better work, i.e. increasing the reward for the winners can stimulate worker's participation, improve the quality of worker's submission and achieve shorter completion time [3, 8-10]. Due to worker's sensitivity to the amount of award, for task requesters, analyzing the impact of pricing strategies is extremely important to make trade-offs among cost saving, degree of participation and expected submission quality [11]. However, there is a lack of consistent evidences on the consequences of different pricing strategies on software worker performance. This motives the study reported here.

In this paper, an empirical study is conducted to investigate the extent to which task pricing strategies are employed in software crowdsourcing, using data extracted from TopCoder (the most popular software crowdsourcing platform) [12]. Meanwhile, we devise an algorithm for impact analysis of pricing strategies on worker behaviors, and obtain some interesting conclusions. The rest of the paper is structured as follows: Section 2 introduces the related literature. Section 3 presents an overview of the research method, including data preparations and experiment descriptions. In section 4, we report the empirical results. Section 5 summarizes discussions and limitations, finally, Section 6 is the conclusion.

2 Related Work

2.1 Pricing Models in Software Crowdsourcing

In terms of pricing issue for crowdsourcing-based software development tasks, Mao et al. [5] built structural as well as non-structural empirical pricing models, based on historical data from TopCoder. They proposed 16 price drivers which fall into four different categories: 1) Development Type (DEV), 2) Quality of Input (QLY), 3) Input Complexity (CPX), and 4) Previous Phase Decision (PRE). In their paper, 12 predictive pricing models derived from 16 pricing drivers were evaluated, including three traditional methods (COCOMO'81, the random guessing, the Naïve model), two regression models (LReg, Logistic), two case-based learners (KNN-1, KNN-k), and five machine learning models (C4.5, CART, QUEST, NNet, SVMR). The study concluded that high predictive quality is achievable, outperforming existing available pricing techniques and providing actionable insights for task requesters.

Additionally, Turki et al. [6] proposed Context-Centric Pricing (CCP) approach by introducing 6 pricing factors extracted from textual task requirements, and investigated 7 different pricing models. Compared with Mao's pricing models, CCP employed only limited information available in the received requirements, however, accuracy did not seem to be improved. As mentioned in previous section, neither models consider additional pricing factors that reflects competitors' status or worker supply levels, which limit to their ability on decision support for making the "right" price.

2.2 Award-Worker Behavior Relationship

Payment plays significantly role on worker's willing to involve in a crowdsourcing task [4-6, 13]. Some studies explored the relationship between task award and worker's behaviors. Table 1 summarizes different viewpoints reported in recent years, which are rather limited and inconsistent, and requires further investigation. On the one hand, several papers reported that higher reward can encourage better work, such as shorter completion time, more positive participation and a higher ability level of winners [2, 3, 8-10]. On the other hand, award seems to play a negative role in worker behaviors in general, i.e. as award increase, the number of registrants, the number of submissions and the quality of the final submission all decrease, although negligible reduction, as shown in [15].

2.3 Pricing Strategies on Supplier Behavior

In the context of traditional demand-supply, buyers usually hope to obtain the lowest possible purchase price for necessary supplies or services. As Michael E. Smith et al. reported in [16], price analysis is a comparative process that seeks to establish reasonable purchase price thresholds relative to market conditions, and the type of market within which the supplier is operating is crucial input into the process. As shown in Fig.2, they investigated pricing strategies in two different types of market, one is competitive marketplace, which contains many suppliers and each small relative to the total size of the market, the other is oligopoly or monopoly market characterized by few relatively powerful producers. Where there is competition, purchasing personnel employs the egocentric pricing strategy, i.e. force down the price by pitting supplier against supplier to achieve purchasing

Table 1. Views on the award-worker behavior relationship

pricing strategy	impact	participation	development time	quality of submission
OVER-PRICE	positive	[3, 8, 10]	[2]	[3, 8, 9, 14]
	negative	[15]	-	[8, 15]
UNDER-PRICE	positive	-	-	-
	negative	-	-	[9]

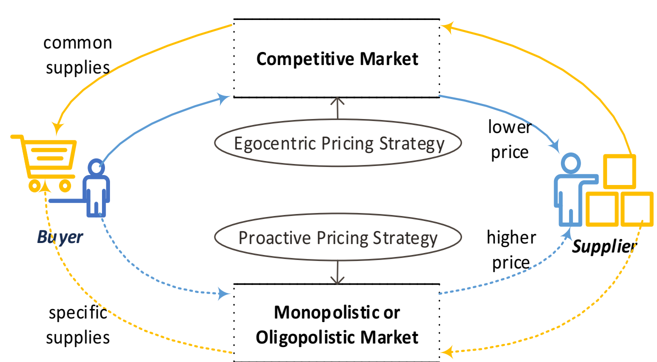

Figure 2. Pricing strategies on supplier behavior

desired products. Furthermore, the competition will be perfect if the product is standardized, whereas it is imperfect when there are many producers of similar product but there is some differentiation. Under oligopolistic or monopolistic condition, purchasers have to employ the proactive pricing strategy, i.e. set a higher purchase price to obtain satisfactory supplies from specific suppliers. In this study, we will adopt similar assumptions on characterizing market conditions and corresponding pricing strategies.

3 Research Method

3.1 Conceptual Pricing Strategies on Worker Behavior

In the context of software crowdsourcing, there is also a demand-supply relationship between task requesters and crowd workers. Correspondingly, requesters are buyers, workers are suppliers, and the marketplace between them is organized by the crowdsourcing platform. Moreover, the pricing issue of crowdsourcing tasks on TopCoder is similar with traditional price analysis of consumer goods. Therefore, we assume that there are also two pricing strategies associated with two different types of market, it is similar with our introduction in section 2.3. Fig. 3 illustrates conceptual pricing strategies on worker behaviors in software crowdsourcing.

Software workers are shared resources in crowdsourcing marketplace. If multiple task requesters release their tasks at the same time, they have to compete on the shared pool of workers by overpricing tasks, to involve enough workers in and obtain desirable worker behaviors. Specifically, in the competitive market, task requesters may need to employ proactive task pricing strategy to achieve good worker performance, e.g. more registrants of task, more submissions, quicker development velocity and better quality of final deliverable.

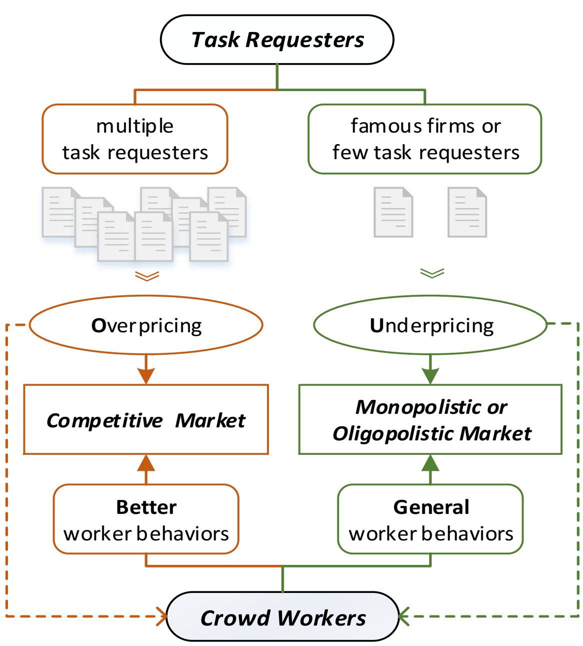

Figure 3. Conceptual pricing strategies on worker behaviors

Additionally, there may also be monopolistic or oligopolistic context although it is rare. For example, workers will register, complete and submit tasks from famous companies such as Google, Facebook, Yahoo, and so on, no matter how low the task award is. As another example, worker's task selection is quite limited, when there are very few requesters. Under such conditions, task requesters may underprice tasks to make trade-offs between cost saving and worker performance, i.e. obtain anticipated worker behaviors at a lower price, it is the egocentric pricing strategy in software crowdsourcing.

3.2 Research Questions

Three research questions are formulated on the above conceptualization. Specifically, it is our interest to analyze the following questions in software crowdsourcing.

RQ1: To what extent over-pricing (i.e. proactive pricing strategies) and under-pricing (i.e. egocentric pricing strategies) were employed in software crowdsourcing practices?

This RQ aims at investigating the distribution of different pricing strategies in software crowdsourcing practices. In order to do so, it is essential to develop methods to derive representative "Nominal" price for each task and use the Nominal price for comparison purpose. The extent to which two pricing strategies (i.e. over-pricing and under-pricing) are employed can then be measured in number of overpriced tasks and underpriced tasks.

RQ2: How to measure the impact of two pricing strategies on worker's behaviors?

This RQ aims at establishing a metric for measuring the impact of different pricing strategies. We propose an ordinal scale metric and develop an algorithm to automatically label certain pricing strategy to be having "negative", "neutral", or "positive" impact on an individual task.

RQ3: What is the consequences of two pricing strategies on worker performance?

This RQ aims to apply the above algorithms and empirically evaluate the consequences of two pricing strategies on the worker's behavior and performance.

3.3 Metrics

We define four metrics to describe worker's behaviors for our analysis, and they are summarized in three categories as shown in Table 2: 1) participation level, including the number of registrants and submissions for the task, i.e. REG and SUB; 2) productivity which is measured by calculating completion velocity, i.e. (time taken/time allowed); 3) quality level which is measured by the score of the winning submission, the score is granted through a peer review performed by experts and experienced developers.

3.4 Dataset

The dataset used was collected by Mao et al. [5], containing 434 crowdsourcing software development tasks from Sep 2003 to Sep 2012. All tasks had received acceptable submissions with score

Table 2. Summary of worker behavior metrics definitions

Category	Metric	Measurement
participant level	REG	number of registrants that are willing to compete on a task. Range: (0,∞)
	SUB	number of submissions before the deadline. Range: (0,∞)
productivity	VELO	velocity=time taken / time allowed, time taken is the number of days actually spent on the task and time allowed is the number of days prescribed in task description. Range: [0,1]
quality level	SCORE	score of the winning submission. Range: (0,100]

Table 3. Summary of metric data in the dataset

Metric	Min	Max	Median	Average	STDEV
Award	112.5	3000	750	752.965	370.357
REG	1	72	16	18.753	11.287
SUB	1	44	4	5.297	5.035
VELO	0	0.875	0.385	0.384	0.162
SCORE	75.01	100	93.86	92.409	6.103

higher than 75, and rewarded the top-2 winners with a 2:1 ratio. In addition, they have complete task completion information. Table 3 shows basic data statistics. Before further analysis, normality test is run on all 4 worker behavior metrics. The results show they do not follow normal distribution. The histogram distribution of REG, SUB and VELO is right skewed, and the distribution of SCORE is left skewed.

3.5 Methodology

To answer the RQs, we conduct an empirical analysis which consists of three phases, as shown in Fig.4. Firstly, we derive a representative Nominal price for each task in the dataset. Secondly, determine whether the task employed pricing strategy or which pricing strategy was employed. Lastly, there is analysis phase, and we will investigate three research questions proposed in section 3.2. As for RQ1, we will complete distribution analysis of pricing strategy in software crowdsourcing, by analyzing the number of overpriced tasks and underpriced tasks. In order to solve RQ2, we design a simple method to measure worker behavior's variations under different pricing strategies, and the details will be introduced in section 4.2. And for RQ3, we try to report the consequences of pricing strategy on worker performance by analyzing worker behaviors' variations.

3.5.1 Derive nominal price

In this step, we apply three different cost modeling approaches to derive a representative "Nominal" price for each task in the dataset, in order to avoid biases associated with individual approach. Three different cost estimation methods are a regression model, an instance-based learner and a machine learning technique: 1) LReg, it is a multiple linear regression model proposed by Mao et al. [5]. 2) 10NN, i.e. 10-Nearest Neighbor, pricing a task with the average award of the 10 nearest neighbors [17]. 3) SVMR, i.e. Support Vector Machine Regression, a machine learning technique [18]. We use 16 price

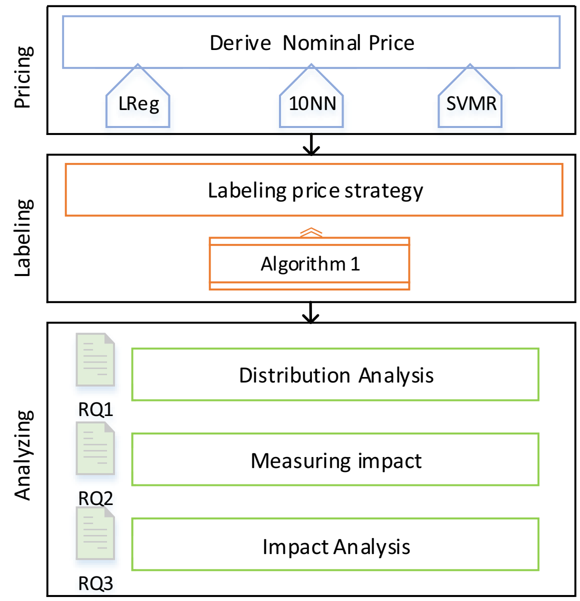

Figure 4. The overview of empirical analysis

drivers proposed by Mao et al. in [5] as input, outputting a drivers proposed by Mao et al. in [5] as input, outputting a "Nominal" price of task by above methods. Elaborated analysis will be carried out in the three scenarios separately.

3.5.2 Labeling price strategy

Then, for each task, we compare the derived Nominal price with its actual price, and label its pricing strategy using one of the three categories, i.e. *over-price*, *under-price* and *Nominal.* The labeling process follows the simplified heuristics: if the Nominal price is lower than actual price, the task will be labelled as "over-price"; if the Nominal price is higher than actual price, the task is labelled as "under-price"; otherwise, the task is labelled as "Nominal", which refers to tasks with actual prices equal to their Nominal prices. Algorithm 1 summarizes the pseudocode for this "Labeling" process.

Algorithm 1 Labeling price strategy

```
% labeling price strategy for each task in the dataset
over-price = Ø
under-price = Ø
Nominal = Ø
for each task t : { t, t∈total dataset } do
    t.N_price <— the Nominal price of t
    t.a_price <— the actual price of t
    if t.N_price - t.a_price < -20
        over-price.add (t)
    else if t.N_price - t.a_price > 20
            under-price.add(t)
        else
            Nominal.add(t)
return over-price, under-price, Nominal
```

3.5.3 Analysis Phase

During this phase, three questions related to pricing strategies in the field of software crowdsourcing will be discussed.

In terms of RQ1, we will count how many tasks are overpriced and how many tasks are underpriced, to analyze the extent to which pricing strategies are employed in software crowdsourcing. As illustrated in Fig. 3, there are two different pricing strategies in crowdsourcing marketplace, i.e. overpricing (the proactive pricing strategy) and underpricing (the egocentric pricing strategy). The number of overpriced tasks and the number of underpriced tasks can simply reflect the distribution of two different pricing strategies.

As for RQ2, an algorithm was proposed to measure the impact of pricing strategies on worker's behaviors. It can label the variation of each worker behavior using one of three labels, i.e. "positive", "negative" and "neutral", and "positive" represents better worker performance, "negative" means that worker behaviors are impacted negatively and "neutral" refers to general worker behaviors.

Regarding RQ3, by comparing the number of "positive" labels and the number of "negative" labels for each worker behavior, we can understand the consequences of pricing strategies on worker performance. The comparison rules as follow: if the number of "positive" is more than "negative", there is a greater chance of obtaining better behavior; if the number of "negative" is greater, corresponding worker behavior is more likely to be influenced negatively.

4 Results

4.1 Answer to *RQ1: To what extent over-pricing (i.e. proactive pricing strategies) and under-pricing (i.e. egocentric pricing strategies) were employed in software crowdsourcing practices?*

In this study, two pricing strategies (proactive and egocentric) are taken into consideration. We start with an analysis of to what extent proactive pricing strategies and egocentric pricing strategies are employed in software crowdsourcing. It is accomplished by running Algorithm 1, in which tasks in the dataset have been labelled as overpriced, underpriced or nominal. Table 4 shows the specific results of Algorithm 1, i.e. the number of overpriced tasks and underpriced tasks in three scenarios. Additionally, corresponding percentages of different pricing strategies are presented. We can learn that more than 33% tasks in the dataset may be overpriced, and at least 39.9% tasks are likely to be underpriced under three scenarios. It indicates that task pricing strategies are prevalent in software crowdsourcing practices. In other words, task requesters usually employ certain pricing strategy to facilitate price decision process.

Answer: In software crowdsourcing, two particular task pricing strategies seem to be prevalent. Using three different pricing models, our analysis shows that on TopCoder platform, averagely about 37.8% tasks may employ the proactive pricing strategy (i.e. over-pricing), and about 46.1% tasks are likely to employ the egocentric pricing strategy (i.e. under-pricing). Only about 16.1% tasks are price similar to the nominal price, implying they seem not to make adjustment for estimated task price.

Algorithm 2 Analysis of pricing strategy

```
% label variations of worker behaviors
for each task t : { t, t∈over-price} OR { t, t∈under-price } do
  find tasks ts with the same t.a_price from the total dataset
  for each metric m : worker behavior metrics =
   {'REG','SUB','VELO','SCORE'} do
     ts.m <— the median m of ts
     t.m <— the m of t
     if t.m > ts.m
       label t with "positive"
     else if t.m < ts.m
          label t with "negative"
        else
          label t with "neutral"
% measure the variation of worker behaviors
for each metric m : worker behavior metrics =
  {'REG', 'SUB', 'VELO', 'SCORE'} do
   m.pos.num = 0
   m.neg.num = 0
   m.neu.num = 0
   for each task t: {t, t∈over-price} OR {t, t∈under-price } do
     if t.m == "positive"
       m.pos.num = m.pos.num + 1
     else if t.m == "negative"
          m.neg,num = m.neg.num + 1
        else
          m.neu,num = m.neu.num + 1
return m.pos.num, m.neg.num, m.neu.num
```

Table 4. Results of labeling price strategy in 3 scenarios

scenario	total data	task categories		
		over-price	*under-price*	*Nominal*
LReg	434	146 (33.6%)	243 (56.0%)	45 (10.4%)
10NN	434	162 (37.3%)	183 (42.2%)	89 (20.5%)
SVMR	434	185 (42.6%)	173 (39.9%)	76 (17.5%)
Average	434	164 (37.8%)	200 (46.1%)	70 (16.1%)

4.2 Answer to *RQ2: How to measure the impact of two pricing strategies on worker's behaviors?*

In this paper, we have devised a method, which is useful to understand the worker behavior's variations in two different pricing strategies, as shown in Algorithm 2. First, we label tasks in *over-price* and *under-price* respectively with "positive" or "negative" or "neutral", based on variations of worker's behaviors. It is worth noting that the task may be labelled with different labels on different behavior metrics, e.g., it is possible that a task is labelled "positive" on REG, however "negative" on the SUB. Next, for all four metrics in a selected tasks set (*over-price* or *under-price*), we count the number of tasks labelled as "positive", "negative" and "neutral" respectively to investigate the variations in different worker behaviors. For example, when the proactive pricing strategy is employed, we would like to believe that over-pricing is more likely to have positive effect on the submission of task, if the number of "positive" is greater than the number of "negative" on SUB.

Answer: Algorithm 2 proposed in this paper describes a simple but effective method to measure variations of interested worker behaviors under proactive pricing strategy or egocentric pricing strategy. It is helpful for task requesters to perceive that there is

Table 5. Variations on 4 worker behaviors of 3 scenarios

		LReg				10NN				SVMR			
subset	**label**	REG	SUB	VELO	SCORE	REG	SUB	VELO	SCORE	REG	SUB	VELO	SCORE
OVER-PRICE	positive	86	79	78	67	90	86	85	80	102	98	103	89
	negative	51	52	65	79	61	62	76	82	72	68	81	96
	neutral	9	15	3	0	11	14	1	0	11	19	1	0
UNDER-PRICE	positive	103	85	113	127	82	65	86	96	79	62	80	86
	negative	130	123	126	116	97	91	91	87	86	90	88	87
	neutral	10	35	4	0	4	27	6	0	8	21	5	0

either positive or negative effect of pricing strategies on worker behaviors. And the method can be easily extended to analyze other worker behaviors except for four behaviors mentioned here.

4.3 Answer to *RQ3: What is the consequences of two pricing strategies on worker performance?*

Algorithm 2 makes it is possible that analyze the effect of pricing strategies on worker behaviors. Table 5 is statistical results of variations on four worker behaviors when employed different pricing strategies in three scenarios (i.e. LReg, 10NN and SVMR). For four worker behaviors: REG, SUB, VELO and SCORE, there are three labels (positive, negative and neutral) may be matched. "Positive" presents that the use of pricing strategy improves the behavior, while "negative" indicates degraded worker behavior. And "neutral" is a behavior label showing basically unchanged. We can clearly observe the number of three labels related to each worker behavior under two pricing strategies from Table 5. Just as envisioned, REG and SUB tend to decrease and VELO is likely to be slowed down, when employing the egocentric pricing strategy (i.e. under-pricing), due to "negative" labels are the most (highlighted with blue in Table 5). For the proactive pricing strategy, the impact on REG, SUB and VELO probably is positive because of dominant "positive" labels (highlighted in yellow), i.e. when overpricing task, REG and SUB is more likely to increase, and VELO tends to be accelerated. These observations imply that task requesters can typically get their extra dollars investment paid-off if employing proactive task pricing strategy.

To our surprise, there are the counter-intuitive outcome on SCORE. Improved SCORE is more likely to be achieved by under-pricing rather than over-pricing. One possible reason is that there are probably less registrants for underpriced tasks. In this context, the worker will be more confident to win, so they are willing to put more effort and expend more days to achieve better deliverable. Whereas, higher award may attract more workers to register the task, making crowd workers perceive broader competition, and undermining their confidence. Hence they would like to try their luck by completing an adequate submission in a shorter time.

Answer: Overall, under-pricing strategy is associated with under-performance in terms of less participation and slow completion velocity, and over-pricing strategy is associated with over-performance in attracting more registrants and submissions, and achieves quicker velocity. However, there appear to be a counter-intuitive outcome on the score of winner, over-pricing seems to do not receive better deliverable while under-pricing brings out improved SCORE. It indicates that overpricing task cannot improve the quality of tasks by simply encouraging more people to participate.

5 Discussions

5.1 Impact of extra incentive on worker participation

Results from RQ3 suggest that there may be positive impact of extra incentive on worker participation. More specifically, over-pricing has a greater chance to attract more workers to register and submit crowdsourcing task, while the registrants and submissions are more likely to decrease under under-pricing condition. These findings are consistent with most existing studies, e.g., Shao et al. [3], Huang et al. [8] and Sun et al. [10], which stated higher award can encourage more positive participation. Moreover, we also provide empirical evidences on the negative impact of lower award on worker participation, which lacks of associated research, as shown in Table 1.

5.2 Impact of extra incentive on worker performance

In this paper, we find that task requesters have already employed pricing strategy to facilitate the price decision process in software crowdsourcing, however, it usually is unconscious. One implication from our results is that extra incentive may encourage better worker performance, i.e. in order to attract broader competition or achieve quicker completion velocity, task requesters can overprice tasks suitably. In other words, the proactive pricing strategy may stimulate better worker behaviors, and the egocentric pricing strategy may lead to general or even declining worker behaviors. This indicates that task requesters can typically get their extra dollars investment paid-off if employing proactive task pricing strategy.

At the same time, what requesters should pay attention to is that underpricing tasks for cost saving tends to decrease the number of registrants and submissions, and lead to a slow velocity. So, they had better to be cautious to under-pricing. Additionally, it is not cost-effective for task requesters to simply raise award for receive better deliverable. Because the quality of final submission seems not to be improved by simply attracting broader competition. To receive better submission, it is recommended to adopt special quality assurance rules, e.g., winners can be rewarded double price if their score is higher than 99.

5.3 Limitations

We are at the beginning of our exploration, a number of limitations should be considered. First, another factor may influence task requesters' pricing strategies is their competitor situations, it is not considered in current empirical study because of data shortage. Second, this study concentrated in TopCoder, the largest crowdsourcing platform. For non-competitive or collaborative crowdsourcing platform, there may be different or specific pricing strategies. Therefore, we cannot claim the generalizability of the results. Third, we proposed a simple method to measure the impact of pricing strategies on worker behaviors. In the future, more correlation analysis and machine learning technologies will be employed in further investigation. Last, workers rating is used in TopCoder, and different ratings represent different levels of skill and experience. It is necessary to analyze the effect of pricing strategies on different workers in the subsequent study.

6 Conclusions

In software crowdsourcing, task requesters may employ different pricing strategies though frequently in unconscious manners, in order to obtain desirable worker behaviors or save cost. Understanding the consequences of different pricing strategies on software worker performance is beneficial to better price decision.

This paper reports an empirical study on task pricing strategy in software crowdsourcing, based on data extracted from TopCoder. There are some interesting findings. In general, under-pricing and over-pricing are prevalent in software crowdsourcing practices, i.e. strategic task pricing patterns have been generally employed in software crowdsourcing. Moreover, task requesters seem to be able to typically get their extra dollars investment paid-off if employing proactive task pricing strategy. We find from the primary statistical analysis that overpricing tasks is more likely to attract more workers to register and submit, and have higher task completion velocity. For underpriced tasks, they tend to associate with less registrants and submissions, and lower task completion velocity. However, it is also observed that it appears to be a counter-intuitive effect on the score of final deliverable. It is possible that workers may exert less effort as overpricing will increase competition, so we can learn that overpricing task cannot improve the quality of final deliverable by simply attracting more workers.

We believe the preliminary findings are helpful for task requesters to facilitate price decision process by understanding the impact of pricing strategies on worker's behaviors, and hope to stimulate further discussions and research in pricing strategies of software crowdsourcing. Future works continuing this study includes: analyze the relationship of competitor's price and task pricing strategy to optimize pricing strategy; conduct an investigation related to whether pricing strategies have similar impact on different types of tasks.

Acknowledgments

The authors would like to thank Ye Yang for her guidance and discussion. We also want to thank the anonymous reviewers for their helpful remarks and comments. This work was supported by the National Natural Science Foundation of China (No.61170312, 61402355, 61502378) and the State Key Laboratory of Software Engineering (No. SKLSE2012-09-14).

References

[1]. K.J. Stol and B. Fitzgerald. Two's company, three's a crowd: a case study of crowdsourcing software development. In *Proceedings of the 36th International Conference on Software Engineering*, 2014.

[2]. S. Faradani, B. Hartmann, and P.G. Ipeirotis. What's the Right Price? Pricing Tasks for Finishing on Time. In *AAAI Conference on Human Computation and Crowdsourcing*, 2011, San Francisco, California.

[3]. B.J. Shao, L. Shi, B. Xu and L. Liu. Factors affecting participation of solvers in crowdsourcing: an empirical study from China. *Electronic Markets*, 22(2): pp. 73-82, 2012.

[4]. Y. Singer and M. Mittal. Pricing mechanisms for crowdsourcing markets. In *International Conference on World Wide Web*, 2013.

[5]. K. Mao, Y. Yang, M.S. Li and M. Harman. Pricing crowdsourcing-based software development tasks, In *the 35th International Conferenece on Software Engineering (ICSE'13)*, San Francisco, CA. pp. 1205-1208, 2013.

[6]. T. Alelyani, K. Mao, and Y. Yang. Context-Centric Pricing: Early Pricing Models for Software Crowdsourcing Tasks. In *The International Conference*, 2017.

[7]. D. Oritsematosan and A.M. Edwin. A Review of The Effect of Pricing Strategies on The Purchase of Consumer Goods. *International Journal of Research in Management Science & Technology*, 2014.

[8]. Y. Huang, P.V. Singh, and T. Mukhopadhyay. Crowdsourcing contests: A dynamic structural model of the impact of incentive structure on solution quality. In *Thirty Third International Conference on Information Systems*, 2012, Orlando.

[9]. G. Kazai, J. Kamps, and N. Milic-Frayling, An analysis of human factors and label accuracy in crowdsourcing relevance judgments. *Information Retrieval*, 16(2): pp. 138-178, 2013.

[10]. Y.Q. Sun, N. Wang, C.X Yin and T. Che. Investigating the non-linear relationships in the expectancy theory: The case of crowdsourcing marketplace. In *the 18th Americas Conference on Information Systems (AMCIS'12)*, 2012.

[11]. J. Wang and P. Ipeirotis, Quality-based Pricing for Crowdsourced Workers. *Social Science Electronic Publishing*, 2013.

[12]. TopCoder. *website: http://www.topcoder.com*.

[13]. D. Dipalantino and M. Vojnovic. Crowdsourcing and all-pay auctions. In *ACM Conference on Electronic Commerce*, 2009.

[14]. M. Yin, Y. Chen, and Y.A. Sun. The effects of performance-contingent financial incentives in online labor markets. In *In 27th AAAI Conference on Artificial Intelligence (AAAI)*, 2013.

[15]. Y. Yang and R. Saremi. Award vs. Worker Behaviors in Competitive Crowdsourcing Tasks. In *ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, 2015.

[16]. M.E. Smith, L. Buddress, and A. Raedels. The Strategic Use of Supplier Price and Cost Analysis. In *91th Annual International Supply Management Conference*, 2006.

[17]. M.E. Bajta. Analogy-Based Software Development Effort Estimation in Global Software Development. In *IEEE International Conference on Global Software Engineering Workshops*, 2015.

[18]. M. Humayun and G. Cui. Estimating Effort in Global Software Development Projects Using Machine Learning Techniques. *International Journal of Information and Education Technology*, 2(3): pp. 208-211, 2012..