Cost-Optimal Validation Mechanisms and Cheat-Detection for Crowdsourcing Platforms

Matthias Hirth, Tobias Hoßfeld, Phuoc Tran-Gia
University of Würzburg, Institute of Computer Science, Germany
Email: [matthias.hirth|tobias.hossfeld|phuoc.trangia]@informatik.uni-wuerzburg.de

***Abstract*—Crowdsourcing is becoming more and more important for commercial purposes. With the growth of crowdsourcing platforms like MTurk or Microworkers, a huge work force and a large knowledge base can be easily accessed and utilized. But due to the anonymity of the workers, they are encouraged to cheat the employers in order to maximize their income. Thus, this paper presents two crowd-based approaches to validate the submitted work. Both approaches are evaluated with regard to their detection quality, their costs and their applicability to different types of typical crowdsourcing tasks.**

***Keywords*-crowdsourcing; cheat-detection mechanism**

I. INTRODUCTION

With the tremendous growth of the Internet's user base, a huge workforce with a large amount of knowledge developed. This is already utilized in projects like Wikipedia, where users created an encyclopedia by sharing their knowledge, or OpenStreetMap which offers maps from all over the world based on information gathered by its users.

A new approach to use this workforce and the wisdom of the crowd is referred to as *crowdsourcing*. Crowdsourcing can be viewed as a further development of outsourcing. Jeff Howe defined crowdsourcing as "... the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call" [1]. The main differences to outsourcing are that the entrepreneur does not know who accomplishes his task and that the workers do not form an organized group but are members of a large anonymous crowd. In traditional work organization, the employer delegates work to the workers, but in the crowdsourcing approach, the worker chooses which tasks he wants to work for.

At the beginning crowdsourcing was often used for non-profit applications. But with the development of platforms like MTurk or Microworkers, which offer an easy access to a huge amount of workers, crowdsourcing became also interesting for commercial usage. Using commercial crowdsourcing work can be done very quickly by accessing a large and relatively cheap workforce, but the results are not reliable. Some workers submit incorrect results in order to maximize their income by completing as many jobs as possible, others just do not work correctly. In the following we denote to all of them as cheaters as the reason for submitting invalid work is irrelevant for our analysis. Sometimes a small amount of incorrect results can be tolerated, but not in general. Therefore, techniques have to be developed to detect cheating workers and invalid work results.

In this paper, we present two approaches to detect cheating workers. As the approaches are based on crowdsourcing, they are easily integrable in common crowdsourcing platforms. We evaluate the quality of our cheat-detection solutions, discuss their costs, and demonstrate their applicability to common crowdsourcing tasks. General guidelines are given how our findings can be used in real crowdsourcing tasks.

The paper is structured as follows. Sec. II gives a quick overview of the concept of crowdsourcing and the research already done in this area. In Sec. III we present our two approaches for work validation, which are evaluated in Sec. IV. The costs of the approaches are analyzed in Sec. V, which also contains relevant use cases for crowdsourcing. The paper is concluded in Sec. VI.

II. BACKGROUND AND RELATED WORK

In the following, we give a quick overview of the general ideas and common terms of crowdsourcing. We show typical examples of crowdsourcing tasks and introduce a rough categorization of these tasks, based on the required worker skills.

A. Crowdsourcing Scheme and Terminology

Every employer needs a mediator to access the worker crowd. This mediator is called *crowdsourcing platform* which is schematically depicted in Fig. 1. Well known examples of these platforms are e.g. MTurk or Microworkers.

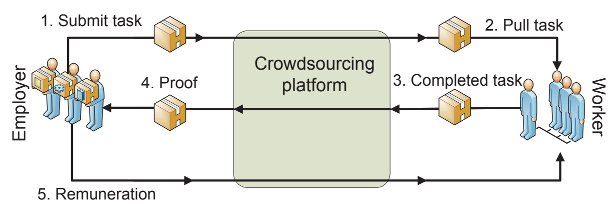

Fig. 1. Crowdsourcing scheme

An employer submits a *task* to the crowdsourcing platform and defines how much the workers will be paid per task and how the workers have to provide *proof* of a completed task. Random workers from the crowd choose to work on the task and after completion submit the required poof to the crowdsourcing platform. The work proof is forwarded to the employer, who pays the worker if the task was completed correctly.

B. Typical Crowdsourcing Tasks and Their Categorization

Crowdsourcing can be used for various purposes which can be roughly categorized into *routine*, *complex*, and *creative tasks*. Routine tasks are jobs that do not require any level of qualification, like bookmarking a web page using social bookmarking services such as digg, relevance evaluation [2], or creating a new YouTube account. *Complex tasks*, like text annotation [3] or rewriting a given text, need some general skills, in contrast to *creative tasks* where highly specialized skills are required. *Creative tasks* include, e.g. writing an article on a given topic or even research and development [4].

Detecting cheating workers is more difficult for complex tasks than for routine tasks. Assume a routine task, where a worker has to create a new YouTube account. The worker has to submit the login data in order to proof that the task is completed. It is easy to check automatically whether the login data is valid or not. This is exemplary for routine tasks, where verification is often simple and easy to automatize. This is different for complex or creative tasks. Assume a complex task, where a worker has to rewrite a given text and a creative task where a worker has to write a text on a given topic. In both cases the worker's texts have to be read and rated according to their content and their style. This can not be automatized and especially for the complex task the reviewer also needs some background knowledge to judge the relevance of the worker's text.

C. Related Work

Crowdsourcing applications suffer from workers, who try to submit an invalid poof in order to receive a payment without completing the required task. The quality of some tasks can be increased by adding verification questions [5] or by using coordination techniques [6]. Furthermore, Eickhoff and Vries [7] showed that depending on the type of task more or less malicious workers are encountered and suggested to derive the quality of a worker not only from the number of completed tasks but also their type, i.e. does the worker only perform simple tasks or mainly complex ones. Chen et al. [8] point out that cheating workers are a problem, but they stated that there is no systematic way to cheat their crowdsourcing platform for quality of experience tests. However, they do not describe general cheat-detection mechanisms. In [9], Ahn and Dabbish present a crowd-based image labeling game, which is now used in an adapted version by Google's Image Labeler. A label is added to the picture, if at least two randomly picked users suggest the same label. Ahn and Dabbish argue that cheating is not possible due to the huge number of players. Two random players are very unlikely to know each other and, hence, are not able to collaborate.

Currently, cheat-detection techniques are either specialized on a certain task type [10], [11], are based on control questions which are evaluated automatically, or rely on manual re-checking by a trustful person. In the next section, we present two generic crowd-based approaches for tasks where control questions are not applicable and manual re-checking is ineffective.

III. Crowd Based Cheat Detection Mechanisms

We propose two crowd-based cheat-detection approaches: A majority decision (MD) and an approach using a control group (CG) to re-checking the main task. In order to analyze these approaches we use the model described in the following.

A. General Notion and Variables

In our model, the crowd consists of N individual workers. Not every worker is honest and performs the task correctly, but we can assume that these workers do not intend to falsify the task result. They only give a random result in order to complete the task as fast as possible. We assume that there is a probability p_c that a randomly chosen worker is a cheater, which means the worker will submit a random result. This result is wrong with a probability of $p_{w|c}$. It is not possible to decide whether a worker submitted a wrong result deliberately or accidentally and the worker is treated as a cheater in both cases. Thus, in our model only cheaters submit incorrect results, i.e. $p_{w|\overline{c}} = 0$. Honest users, who accidentally submit an invalid result can be modeled by adjusting p_c. Therefore, the probability of a wrong task result is $p_w = p_c \cdot p_{w|c}$. To clarify this imagine a multiple choice test with one correct answer out of five possibilities and a crowd of 100 workers including 10 cheaters. The probability of choosing a cheater is $p_c = 10\%$, the probability for picking a wrong answer when choosing randomly $p_{w|c} = 80\%$. This results in a probability for a wrong answer $p_w = 8\%$.

B. The Majority Decision Approach

Our first approach (MD) uses a majority decision to eliminate incorrect results and is illustrated in Fig. 2. The employer submits his task to the crowdsourcing platform (Step 1 in Fig. 2). The platform duplicates the task (2) and i different workers complete the tasks. They submit their individual results (3) which might be correct or incorrect. The crowdsourcing platform performs a majority decision (4), i.e., the result most of the workers submitted is assumed to be correct and returned to the employer (5). In this approach each worker submitting a result is paid.

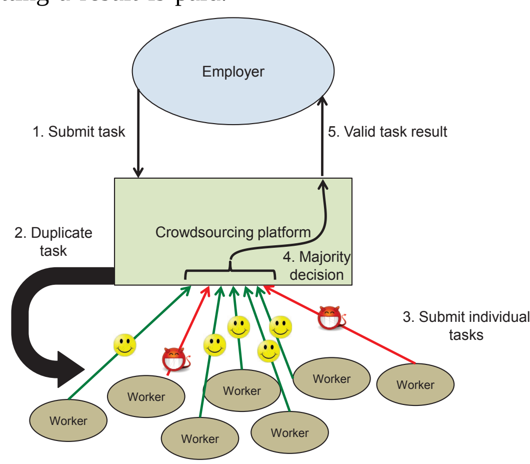

Fig. 2. *Majority Decision* (MD) approach scheme

As an example application of the MD approach, think of 100 workers searching for relevant web pages to a given topic. If a one web page is submitted by 92 workers, it is certainly relevant for the given topic. Even if some workers are cheating, the overall result is valid.

C. The Control Group Approach

Our second approach (CG) is based on a control group and is schematically depicted in Fig. 3. The employer submits the

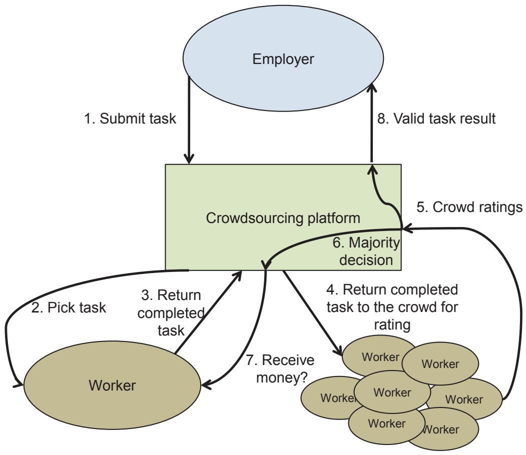

Fig. 3. *Control Group* (CG) approach scheme

main task to the crowdsourcing platform (1) and the task is chosen by a worker (2), who submits the required task result (3). Now, the crowdsourcing platform generates new validation tasks for this result. The result of the main task is given to a group of j other workers, who rate it according to given criteria (4). The ratings of the different workers are returned to the crowdsourcing platform (5), which calculates the overall rating of the main task (6). The main task is considered to be valid, if the majority of the control group decides the task is correctly done. This is necessary, because some workers in the control group may be cheating and submit wrong ratings. If the main task is rated valid, the main worker is paid (7) and the result is returned to the employer (8). An important point of this approach is that the main task and the ”re-check” task are assumed to have different costs. Usually, the main task is expensive, while the control task is cheaper.

An application of this approach is a task where a worker has to write an article including certain keywords. This is a creative task and thus expensive. The completed article is given to 100 low paid workers who have to judge whether the article matches the initial keywords or not. If enough workers submit that the article matches, it is assumed to be valid.

IV. Evaluation of the MD and the CG approach

Both approaches use a majority decision of m workers in order to verify the task result. Thus, we have a closer look at how to optimize m, in order to minimize the costs and maximize the reliability of the results. Afterwards, we evaluate the quality of the cheat-detection of both approaches.

A. Group Size for Majority Decisions

For a majority decision we use m random workers from the total crowd of N workers. Each of their m results is with a probability of p_w incorrect and thus, the number of incorrect results X follows a binomial distribution $X \sim \mathrm{BINOM}(m, p_w)$. To derive a correct majority decision, the number of incorrect results has to be smaller than $m/2$, i.e., the probability of a correct majority decision p_m is given by

$$p_m = P\left(X < \tfrac{m}{2}\right) = \sum_{k=0}^{\left\lfloor \frac{m-1}{2} \right\rfloor} \binom{m}{k} p_w^k (1-p_w)^{m-k}. \qquad (1)$$

The probability of a correct majority decision p_m is not constantly increasing with the group size, but is dependent on the parity of the group. Smaller groups of an odd parity always achieve better results than slightly larger groups of an even parity, a mathematical poof is given in [12]. The exploitation of this effect is useful in any application of majority decisions, as it improves the reliability of the results and to lowers the costs.

We analyze the possible cost savings in more detail. To this end, we calculate how many workers are needed for the 99% quantile of a correct majority decision (1) using a group with even parity and (2) using a group with odd parity. For both groups the number of workers is dependent on p_w. The results are depicted in Fig. 4. Note that the y-axis is in logarithmic scale. The required size of the even group is shown by the dark continuous line, the size of the odd group by the dashed line and the difference of both group sizes by the light continuous line.

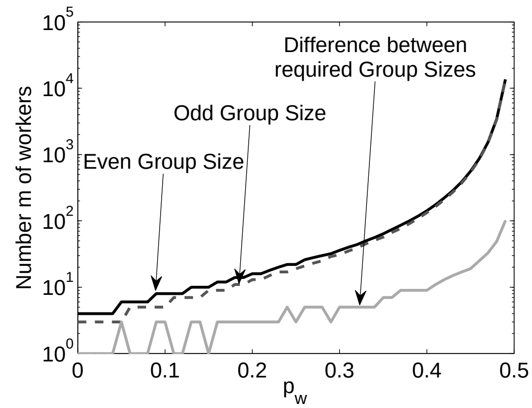

Fig. 4. 99% Quantile of the required number m of workers for a correct majority decision depending on the probability p_w of a wrong task result

From Fig. 4 we can conclude two findings. First, we can at least save one worker when using an odd instead of an even group size without reducing the quality of the majority decision. Second, the higher p_w is the more workers can be saved using an odd group size. In the remainder of this paper we will only use odd group sizes for majority decisions.

B. Quality Comparison of MD and CG Approach

One of the main questions of this work is, whether the MD or the CG approach gives better results in terms of cheat-detection quality. In order to compare both approaches, we use the same number of workers m for the MD approach and for the control group of the CG approach, i.e. $i = j = m$.

1) MD Approach: Having a look at the MD approach, we only have to evaluate whether the group made a correct or an incorrect decision. The probability for a correct result using the MD approach p_{MD} is the same as the one given in Eq. 1. Thus,

$$p_{\mathrm{MD}} = p_m. \qquad (2)$$

The probability for an incorrect MD result $\overline{p_{\mathrm{MD}}}$ is given by

$$\overline{p_{\mathrm{MD}}} = 1 - p_m. \qquad (3)$$

2) CG Approach: The CG approach is more complicated. We have to differentiate as the main worker and the control group may try to cheat. This results in four possible cases. To describe these cases, we introduce the notation: p_{ab} with

$$a = \begin{cases} r, & \text{worker submits right result} \\ w, & \text{worker submits a wrong result} \end{cases}$$

$$b = \begin{cases} d, & \text{control group makes a right decision} \\ \overline{d}, & \text{control group makes a wrong decision} \end{cases}$$

We assume that our crowd is very large, thus the main worker and the workers from the control group do not know each other. Further, the main task and the control tasks are very different tasks, as the main task is a complex one and the control tasks are rather simple. Hence, the cheating probabilities of the main worker and the control group are independent. Thus, the possible results of the CG approach are:

(C.1) Worker submits a wrong result and the control group decides the result is invalid: $p_{wd} = p_w \cdot p_m$

(C.2) Worker submits a wrong result and the control group decides the result is valid: $p_{w\overline{d}} = p_w \cdot (1 - p_m)$

(C.3) Worker submits a right result and the control group decides the result is valid: $p_{rd} = (1 - p_w) \cdot p_m$

(C.4) Worker submits a right result and the control group decides the result is invalid: $p_{r\overline{d}} = (1 - p_w) \cdot (1 - p_m)$

The probability for a correct result using the CG approach p_{CG} is hence given by

$$p_{\text{CG}} = p_{wd} + p_{rd} = p_m, \tag{4}$$

and the probability for an incorrect result using the CG approach $\overline{p_{\text{CG}}}$ by

$$\overline{p_{\text{CG}}} = p_{w\overline{d}} + p_{r\overline{d}} = 1 - p_m. \tag{5}$$

Comparing p_{MD} and p_{CG}, we can see that the both the MD and the CG approach offer the same quality of cheat-detection quality:

$$p_{\text{MD}} = p_{\text{CG}} = p_m \tag{6}$$

But they differer among there applicability for different crowdsourcing tasks and their costs, as shown in the next section.

V. Application of a Cost Model for different Use Cases

Before we give use cases for each of the control approaches, we specify a cost model. As the presented techniques are intended to be used in real crowdsourcing applications, the economic aspect is not negligible and has to be considered.

A. Cost Model

Crowdsourcing workers are only paid if the submitted result is valid. Therefore, we define costs only for a successfully completed task. We denote these costs as c_1. For the control task in the CG approach, we assume different costs $c_2 \leq c_1$. Approving an invalid task does not only waste money, but has further negative impacts, like encouraging workers to continue cheating or reputation loss. To account for these negative effects, we introduce costs c_{fp} for a "false-positive approval", if an invalid task is not detected. Not paying for correct work has negative influences, too, as workers stop working for this employer. Hence, we use a penalty c_{fn} for a "false-negative approval", if a correct task is assumed to be invalid.

We now calculate the expected cost for both approaches. We use $i = j = m$ workers, thus, the probability for a correct MD and CG approach result is p_m. This analysis helps employers to decide, which approach is cheaper for a certain use cases.

1) MD approach: There is no validation of the individual results in the MD approach, hence every worker is paid c_1 and a false-negative approval can not occur. If the MD result is wrong, the costs are increased by c_{fp}. This results in the total expected costs c_{MD},

$$c_{\text{MD}} = c_1 \cdot m + p_w \cdot c_{fp}. \tag{7}$$

2) CG approach: In the CG approach one worker is working on the main task, which costs c_1 if the worker completed it successfully. The main task is controlled by m workers. Each of the m workers is paid c_2 as these tasks are not validated. The cost of the CG approach varies, whether the worker on the main task is cheating or not, and whether the control crowd judges the result of the main task correctly. To calculate the total expected cost c_{CG} we have to consider the costs of four cases.

(C.1) $c_{wd} = m \cdot c_2$

(C.2) $c_{w\overline{d}} = c_1 + m \cdot c_2 + c_{fp}$

(C.3) $c_{rd} = c_1 + m \cdot c_2$

(C.4) $c_{r\overline{d}} = m \cdot c_2 + c_{fn}$

Therefore, c_{CG} is given by

$$c_{\text{CG}} = c_{wd} \cdot p_{wd} + c_{w\overline{d}} \cdot p_{w\overline{d}} + c_{rd} \cdot p_{rd} + c_{r\overline{d}} \cdot p_{r\overline{d}} \tag{8}$$

with the probabilities calculated in Sec. IV-B2

We now have a look at typical use cases of crowdsourcing. In the following we consider which approach, i.e. MD or CG, is optimal in terms of costs for which kind of crowdsourcing task, i.e. routine, complex and creative tasks.

It has to be noted that the costs assumed in the following sections are typical values which are taken from a large crowdsourcing platform. The costs are normalized to $c = 1$ which is the lowest payment in the crowdsourcing platform. Details can be found in the technical report [12].

B. Routine Task

Routine tasks are typically low paid with $c_1 = 1$ for the main task. The task of re-checking the main task should not be higher paid, thus, we pay $c_2 = 1$ for the control task. The costs caused by a cheating worker are very low in this case. Usually it does not matter if one of the workers does not fulfill the task, but as he might be encouraged to continue cheating we impose a penalty for each approval of an invalid task of $c_{fp} = 1$. Refusing to pay a worker who completed his task, will stop him from working for this employer. But as the crowd contains many worker who can complete simple tasks, the penalty $c_{fn} = 1$ is low.

The resulting costs depending on p_w are displayed in Fig. 5. The costs of the MD approach c_{MD} are shown by the continuous line, the costs for the CG approach c_{CG} by the dashed line. c_{MD} follows a linear growth as it is based on the fixed costs of the majority decision with an additional penalty for a false positive approval, which becomes more likely as p_w increases. The development of c_{CG} is more complex. At $p_w = 0$, the main worker surely earns $c_1 = 1$ as he is not cheating and the control group earns $5 \cdot c_2 = 5$, which results in $c_{\text{CG}} = 6$. With increasing p_w, the main worker is more likely to cheat, whereby c_{CG} decrease, as the control group detects this and the worker will not be paid. With p_w increasing further,

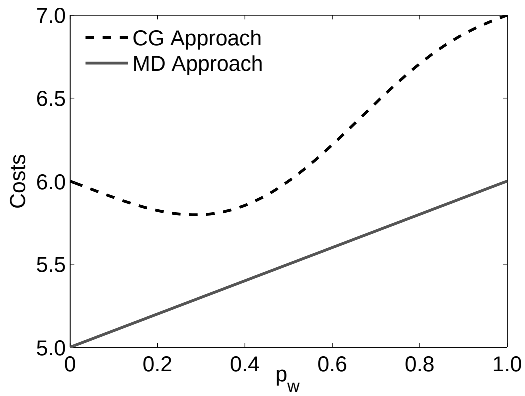

Fig. 5. Costs of an unqualified task dependent on p_w

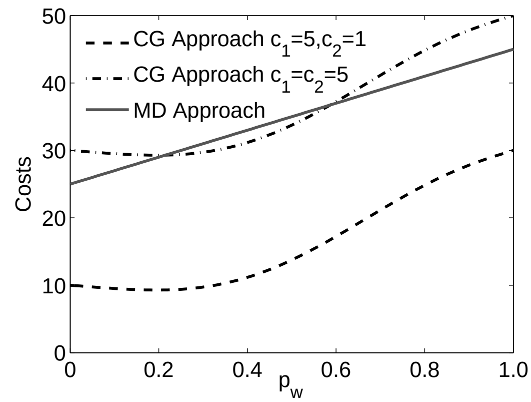

Fig. 6. Costs of a qualified task dependent on p_w

the worker is even more likely to cheat, but also the quality of the control group decreases. The growing number of incorrect decisions leads to increasing costs, due to c_{fp}.

However, in the case of an unqualified task, the costs of the CG approach are always higher than the costs of the MD approach, since generally $c_2 = c_1$ and $c_{pf} \approx c_1$. Thus, the MD approach should be preferred for routine tasks.

C. Complex and Creative Task

For complex and creative tasks wrong results usually have a large impact on the employer. Assume an advertisement campaign in forums. The employer wants to promote a product in web forums dealing with topics related to the product. Direct advertisement is not desired in forum posts, hence, the advertisement has to be hidden in a normal post using a recommendation which fits in the context of a forum thread. As the worker has to find an appropriate forum thread and writes an individual text, we assume $c_1 = 5$ in this case. As proof the worker submits a link to the thread where he posted the advertisement. The control group checks the given thread, if the post is related to the topic and includes the desired recommendation. Each member of the control group is paid $c_2 = 1$. In order to study the influence of c_2 we also calculate the costs for $c_2 = c_1 = 5$. If the advertisement campaign is recognized by the forum administrators, the posts are deleted and a negative discussion about the employer will arise. Thus, the penalty for approving wrong posts is set very high to $c_{fp} = 20$. Besides this, qualified workers for the main task are rare and losing one of them is not desirable. Because of this, we assume $c_{fn} = 5$.

The resulting costs are depicted in Fig. 6, with c_{MD} shown by the continuous line and c_{CG} by the dashed line. The costs curves of c_{MD} and c_{CG} show a similar shape to that for the unqualified task. However, in this case the CG approach with $c_1 = 5$ and $c_2 = 1$ is always cheaper than the MD approach, because of the high penalty for false positive approvals and the low costs for the control task $c_2 < c_1$. If the costs for the control task are raised $c_2 = c_1 = 5$, the CG approach only performs better than MD for $p_w \in [0.20, 0.58]$.

We can derive two guidelines for complex and creative tasks. If the cost ratio $c_2/c_1 \ll 1$, the CG approach should be favored. Otherwise, the cost optimal cheat-detection mechanism can be found by applying our cost model.

D. Cost-Quality Optimization Guidelines for Complex and Creative Tasks

A cost-quality optimization,i.e. finding a trade-off between cheat-detection quality and the costs, for complex and creative tasks is important as they are expensive compared to routine tasks. For this kind of task, the CG approach outperforms the MD approach in terms of costs in most cases. Hence, we will focus on the CG approach in the following.

1) Optimizing Overall Costs and Cheat-Detection Quality: In order to reduce the total costs c_{CG}, a smaller control crowd can be used. But this negatively affects the quality of the cheat-detection, as p_{CG} decreases with the group size. Though, a trade-off between c_{CG} and p_{CG} exists. For our evaluation we use the example of the forum advertisement campaign with $c_1 = 5, c_2 = 1, c_{fp} = 20$ and $c_{fn} = 5$. Fig. 7 depicts c_{CG} depending on p_{CG} for different values of p_w. As c_{CG} remains almost constant for $p_{\text{CG}} < 0.5$, we focus only on $p_{\text{CG}} \geq 0.5$.

Fig. 7 shows that c_{CG} increases with p_w and p_{CG}. A better cheat-detection quality p_{CG} needs more workers leading to higher costs. Also with an increase of p_w more workers are required to achieve a valid result. For a small value of p_w the influence of p_{CG} on the costs is only marginal and increasing the cheat-detection quality is rather cheap. For high values of p_w the costs increase tremendously with p_{CG}, which makes an detection improvement extremely expensive.

We can assume that an employer can approximately determine p_w based on the results of previous tasks. Hence, this model allows him to make a trade-off between costs and result quality according to his needs, by calculating the required m.

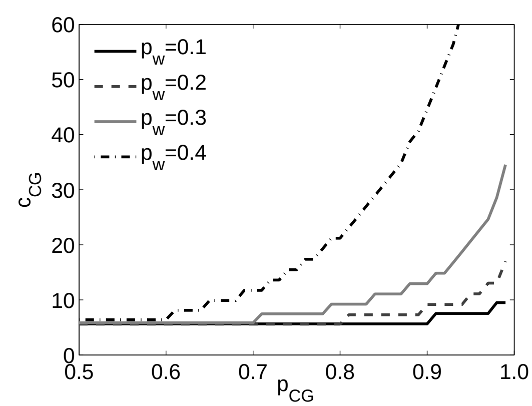

Fig. 7. Total costs c_{CG} depending the cheat-detection quality p_{CG} for different probabilites for wrong answers p_w

To illustrate this, we have a look at two examples with $p_w = 0.4$. Assume an employer wants to spend $c_{\text{CG}} = 30$ for the campaign. We can derive from Fig. 7 that p_{CG} will be about 84%. For $p_w = 0.4$ Eq. 1 can be solved numerically to $m = \frac{ln(1-p_{\text{CG}})+1.0404}{-0.0310}$ and we can calculate the required control group size $m = 25$. The second use case is an employer who demands $p_{\text{CG}} = 90\%$ for his campaign. We can calcuate the required group size $m = 40$ and derive $c_{\text{CG}} = 45$ from Fig. 7.

2) Maximizing Available Salary for the Main Task: Creative and complex tasks require special skills. In order to attract skilled workers, these tasks are better paid than routine tasks. But the costs c_{CG} for a task using the CG approach are split between the main task worker and the control group workers. Thus, we have to make a trade-off between the available money for the main worker and the cheat-detection quality.

We calculate the overhead costs $c_{\text{CG-overhead}}$ using Eq. 8 and setting $c_1 = 0$ dependent on the desired p_{CG}. For a fixed budget b, the maximal available salary c_1 can now be calculated by,

$$c_1 = b - c_{\text{CG-overhead}}.$$

We introduce the quotient $\varepsilon = c_1/b$ as a measure for the efficiency of the cost distribution. $\varepsilon = 1$ means that the entire budget is spent for the main task. Fig. 8 depicts ε for different budgets b and different cheat-detection qualities p_{CG}.

The intersection of the curves and the x-axis mark the minimum required task budget for the given p_{CG}. At this intersection point, no salary for the main task is available. With increasing budget b, more salary for the main task is available as $c_{\text{CG-overhead}}$ remains constant. For large budgets, the main task salary is the biggest part b. The intersections of the curves and the x-axis move to the right for higher p_{CG}, which shows that the higher the desired cheat-detection quality the more expensive the task. With higher p_{CG} also the efficiency of the cost distribution degrades quickly and large amount of the budget is spent on the control crowd instead of the main worker. Therefore, an employer has to consider carefully the required p_{CG}.

VI. Conclusion

Crowdsourcing has only recently developed, but due to its various applications it is becoming an important new form of work organization. One of the major problems are untrustworthy workers trying to maximize their income by submitting as many tasks as possible even if they did not complete the task.

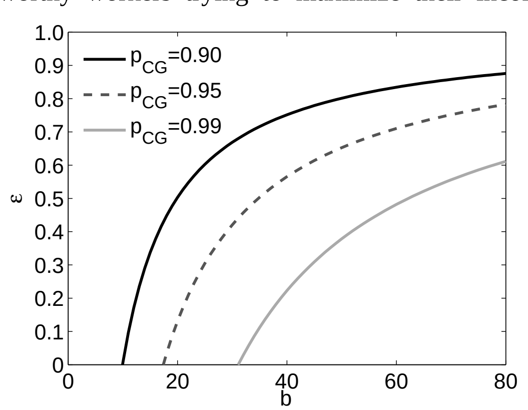

Fig. 8. Efficiency of the cost distribution ε depending on the budget b

As manual re-checking of each task is not desirable, we proposed two different crowd-based methods, the MD and the CG approach, to verify task results. We have shown that, using the same amount of workers, both approaches offer the same significance level for detecting cheating workers. Furthermore, we have proven that it is generally better to use an odd group size instead of an even group size for the presented approaches. Using this finding helps to improve the quality and to reduce the costs of majority decisions.

The costs of the MD and the CG approach were analyzed using typical types of crowdsourcing tasks. This analysis revealed that the MD approach is more suitable for low paid routine tasks, whereas the CG approach performs better for high priced tasks. In order to minimize the costs of high priced tasks, the CG approach was investigated in more detail. We showed that a slight reduce of the cheat-detection quality can significantly lower the cost for the whole task. Similarly, the overhead costs of the CG approach can be significantly decreased by slightly decreasing the cheat-detection quality.

Our approaches showed that crowd-based cheat-detection mechanisms are cheap, reliable, and easy to implement. They help to reduce the cost and the time consumption currently imposed by the manual validation process of task results.

Acknowledgment

This work was conducted within the Internet Research Center (IRC) at the University of Würzburg. The authors alone are responsible for the content of the paper.

References

[1] J. Howe. (2006, Jun.) The Rise of Crowdsourcing. [Online]. Available: http://www.wired.com/wired/archive/14.06/crowds.html

[2] O. Alonso, D. E. Rose, and B. Stewart, "Crowdsourcing for Relevance Evaluation," *SIGIR Forum*, vol. 42, no. 2, 2008.

[3] P. Hsueh, P. Melville, and V. Sindhwani, "Data Quality from Crowdsourcing: A Study of Annotation Selection Criteria," in *Proceedings of the NAACL HLT Workshop on Active Learning for Natural Language Processing*, Boulder, Colorado, USA, May 2009.

[4] InnoCentive, Inc., "InnoCentive," www.innocentive.com [Accessed Mar. 07, 2011].

[5] A. Kittur, E. H. Chi, and B. Suh, "Crowdsourcing User Studies with Mechanical Turk," in *Proceeding of the ACM SIGCHI Conference on Human Factors in Computing Systems*, Florence, Italy, Apr. 2008.

[6] A. Kittur and R. E. Kraut, "Harnessing the Wisdom of Crowds in Wikipedia: Quality through Coordination," in *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, San Diego, California, USA, Nov. 2008.

[7] C. Eickhoff and A. de Vries, "How Crowdsourcable is Your Task?" in *Proceedings of the ACM WSDM Workshop on Crowdsourcing for Search and Data Mining*, Hong Kong, China, Feb. 2011.

[8] K. Chen, C. Chang, C. Wu, Y. Chang, C. Lei, and C. Sinica, "Quadrant of Euphoria: A Crowdsourcing Platform for QoE Assessment," *IEEE Network*, vol. 24, no. 2, 2010.

[9] L. Von Ahn and L. Dabbish, "Labeling Images with a Computer Game," in *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, Vienna, Austria, Apr. 2004.

[10] T. Moore and R. Clayton, "Evaluating the Wisdom of Crowds in assessing Phishing Websites," *Financial Cryptography and Data Security*, 2009.

[11] P. G. Ipeirotis, F. Provost, and J. Wang, "Quality Management on Amazon Mechanical Turk," in *Proceedings of the ACM SIGKDD Workshop on Human Computation*, Washington, DC, USA, Jul. 2010.

[12] M. Hirth, T. Hoßfeld, and P. Tran-Gia, "Cheat-Detection Mechanisms for Crowdsourcing," University of Würzburg, Tech. Rep. 474, 2010.