Leveraging Crowdsourcing to Detect Improper Tasks in Crowdsourcing Marketplaces

Yukino Baba and **Hisashi Kashima**
The University of Tokyo

Kei Kinoshita and **Goushi Yamaguchi** and **Yosuke Akiyoshi**
Lancers Inc.

Abstract

Controlling the quality of tasks is a major challenge in crowdsourcing marketplaces. Most of the existing crowdsourcing services prohibit requesters from posting illegal or objectionable tasks. Operators in the marketplaces have to monitor the tasks continuously to find such improper tasks; however, it is too expensive to manually investigate each task. In this paper, we present the reports of our trial study on automatic detection of improper tasks to support the monitoring of activities by marketplace operators. We perform experiments using real task data from a commercial crowdsourcing marketplace and show that the classifier trained by the operator judgments achieves high accuracy in detecting improper tasks. In addition, to reduce the annotation costs of the operator and improve the classification accuracy, we consider the use of crowdsourcing for task annotation. We hire a group of crowdsourcing (non-expert) workers to monitor posted tasks, and incorporate their judgments into the training data of the classifier. By applying quality control techniques to handle the variability in worker reliability, our results show that the use of non-expert judgments by crowdsourcing workers in combination with expert judgments improves the accuracy of detecting improper crowdsourcing tasks.

1 Introduction

Crowdsourcing platforms provide online marketplaces for outsourcing various kinds of tasks to a large group of people. With the recent expansion of crowdsourcing platforms such as Amazon Mechanical Turk (MTurk)[1] and CrowdFlower,[2] the idea of crowdsourcing has been successfully applied in various areas of computer science research, including computer vision (Sorokin and Forsyth 2008) and natural language processing (Snow et al. 2008). Business organizations also make use of crowdsourcing for processing a large number of tedious tasks such as transcription and product categorization (Ipeirotis 2010).

One of the biggest challenges in crowdsourcing is ensuring the quality of the results submitted by crowdsourcing workers, because there is no guarantee that all workers have sufficient abilities needed to complete the offered tasks. Some faithless workers also try to get paid as easily as possible by submitting worthless responses. Several approaches geared toward efficient quality control have been applied; for example, MTurk provides a pre-qualification system to assess a prospective skill level of worker beforehand, and CrowdFlower enables requesters to inject gold standard data, that is, a collection of tasks with known correct answers, into their tasks to measure a worker's performance automatically. Another promising approach is to introduce redundancy, which asks multiple workers to work on each task, and then aggregates their results to obtain a more reliable result by applying majority voting or other sophisticated statistical techniques (Dawid and Skene 1979; Whitehill et al. 2009; Welinder et al. 2010).

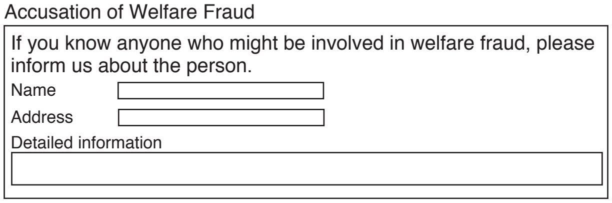
Controlling the *quality of tasks* is another big challenge especially in crowdsourcing marketplaces. For maintaining the safety and integrity of the marketplaces, administrators in the marketplaces have to prevent requesters from posting illegal or objectionable tasks, and to remove improper tasks immediately to prevent the workers from working on them. Most existing crowdsourcing marketplaces prohibit specific kinds of tasks, for examples, ones entailing illegal or antisocial activities, ones collecting personal identifiable information of workers (Figure 1), or ones requiring workers to register for a particular service (Figure 2). Operators in crowdsourcing marketplaces have to monitor the tasks continuously to find such improper tasks; however, manual investigation of each task is too expensive.

In this paper, we present the reports of our trial study on automatic detection of improper tasks, which we conducted on Lancers[3], a popular crowdsourcing marketplace in Japan. In order to support the monitoring of activities by marketplace operators, we apply machine learning to this problem. Our proposed procedure to reduce the monitoring burden comprises three steps. (1) The operators annotate a portion of tasks to indicate whether each task is proper, and train a classifier by applying a supervised machine learning method to the annotated tasks. (2) When a new task is posted, the trained classifier determines whether the task is proper. (3) If the classifier finds potentially improper tasks, the monitoring system reports them to the operator for a manual judgment.

[1] https://www.mturk.com/

[2] http://crowdflower.com

[3] http://www.lancers.jp



Accusation of Welfare Fraud

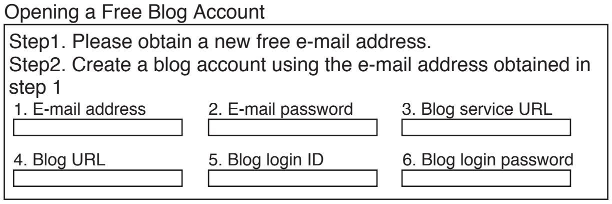
If you know anyone who might be involved in welfare fraud, please inform us about the person.

Name

Address

Detailed information

Figure 1: Example of an improper task (requiring disclosure of another person's identity)



Opening a Free Blog Account

Step1. Please obtain a new free e-mail address.
Step2. Create a blog account using the e-mail address obtained in step 1

1. E-mail address	2. E-mail password	3. Blog service URL
4. Blog URL	5. Blog login ID	6. Blog login password

Figure 2: Example of an improper task (requiring registration at another web service)

To reduce the annotation costs further and improve the classification accuracy, we consider the use of crowdsourcing for task annotation. We hire a set of crowdsourcing workers to monitor posted tasks, and incorporate their judgments into the training of the classifier. Since the crowdsourcing workers are not experts in judging task impropriety, the quality of worker judgments is often lower than that of operators, and moreover, the reliability of their judgments varies significantly depending on workers. Such variability motivates us to resort to applying quality control techniques to create accurate classifiers.

In this paper, we perform a feasibility study of our approach by using real task data from the Lancers crowdsourcing marketplace. We first show that the classifier trained by the expert judgments achieves high accuracy (0.950 averaged area under the curve (AUC)) in detecting improper tasks. We also collect judgments from the crowdsourcing workers of Lancers, and train a classifier using the judgments of both experts and workers. Our results show that incorporating the judgments of crowdsourcing workers achieves a statistically significant improvement (0.962 averaged AUC), and the use of crowdsourced labels allows a reduction in the number of expert judges by 25% while maintaining the level of detection performance.

In summary, this paper makes three main contributions:

1. To the best of our knowledge, our work is the first to investigate the real operational data inside a commercial crowdsourcing marketplace, and to address the issue of task quality control problem in crowdsourcing.
2. We apply a machine learning approach to the task quality control problem and show that the machine learning approach is highly effective in detecting improper tasks in a real crowdsourcing marketplace (Section 3).
3. We show that the use of non-expert judgments by crowdsourcing workers in combination with expert judgments improves the accuracy of detecting improper crowdsourcing tasks (Sections 4 and 5).

2 Improper task detection in crowdsourcing marketplaces

2.1 Improper task detection problem

Our goal is to construct a classifier for detecting improper tasks. We formulate this problem as a supervised machine learning problem. Let us assume there are N crowdsourcing tasks, and each task is represented as a D-dimensional real-valued feature vector denoted by $\boldsymbol{x}_i \in \mathbb{R}^D$. Crowdsourcing marketplaces have their individual definitions of improper tasks, and the operators (i.e., domain experts) give judgments for the tasks. Let us denote the expert judgments for a task i by $y_{i,0} \in \{0, 1\}$, where a label 1 indicates an improper task and 0 indicates otherwise. In addition to the experts, J crowdsourcing workers are requested to annotate the tasks, and we denote a set of workers who give judgments on task i by $\mathcal{J}_i \subseteq \{1, 2, \cdots, J\}$. Note that each worker is not required to annotate all the tasks. Let $y_{i,j} \in \{0, 1\}$ be the annotation on task i by worker j.

Our goal is to estimate an accurate binary classifier $f : \mathbb{R}^D \to \{0, 1\}$ given the annotated dataset $(\{\boldsymbol{x}_i\}_{i \in \{1,2,\cdots,N\}}, \{y_{i,j}\}_{i \in \{1,2,\cdots,N\}, j \in \mathcal{J}_i}, \{y_{i,0}\}_{i \in \{1,2,\cdots,N\}})$ as a training dataset.

2.2 Dataset

We collected task data posted on a commercial crowdsourcing marketplace, Lancers, from June to November 2012, and created a dataset consisting of 96 improper tasks (judged by the operators) and $2,904$ randomly selected proper tasks.

To simulate the task monitoring by crowdsourcing workers, we hired a set of workers on Lancers and requested them to examine each tasks. We generated batches of 15 tasks, and each worker was asked to review a single batch at a time. Note that we did not apply any strategy for worker selection. Each task was examined by two or three workers. General statistics of our collected annotations of workers is given in Table 1.

3 Training Classifier with Expert Judgments

We construct a classifier using only expert judgments to verify the effectiveness of improper task detection by using machine learning. This section presents the details of the features we used for training and the results of the evaluation using the dataset from actual crowdsourcing tasks.

3.1 Features

We prepared three feature types, namely, textual task feature, non-textual task feature, and non-textual requester feature. There is an assumption that motivates us to use requester features as well as task features that there could be specific patterns of requesters likely to post improper tasks.

- **Textual task feature**

  To implement textual task features, we use a simple bag-

Table 1: Statistics about the datasets of workers judgments

#all tasks	#improper tasks	#total judgments by workers	Avg. #judgments per task	#all workers	Avg. #judgments per worker	Total amount of payment ($)
3000	96	8990	2.997	97	92.68	107.4

of-words representation of terms in the task title, description and instruction with binary term frequencies. We ignore the symbols and numbers. We also drop the terms appearing in only one task.

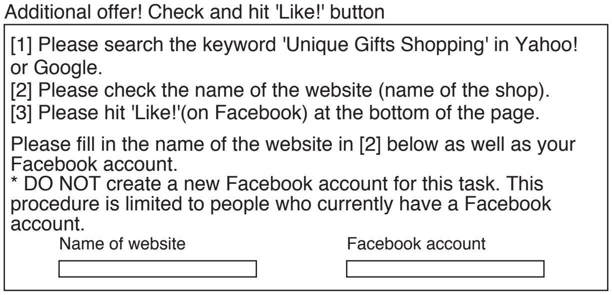
- **Non-textual task feature**
  These features describe the properties of a task such as the number of batches in a series of tasks, amount of payment, the number of workers assigned to the same task, and criteria of worker filtering. By considering these features, we attempt to capture information not represented in the textual features.
- **Non-textual requester feature**
  Information of "Who posts the task?" and "What kind of person posts the task?" may helpful in detecting improper tasks. We consider the following information: the ID of a task requester, requester profiles (gender, year of birth, geo location, and occupation), and trustworthiness of requesters (status of identification and reputation from workers).

3.2 Results

We extracted the entire features presented in the previous session from the dataset we prepared in Section 2.2 and trained a classifier using them. We use 60% (1,800) of the tasks for training and the remaining for the test. We used linear support vector machine (SVM) implemented in LIBLINEAR[4] as a classification algorithm. We evaluated the detection performance with the average and the standard deviation of the AUC over 100 iterations. The results are shown in Table 2. The classifier achieves 0.950 for averaged AUC; therefore, we could confirm that machine learning is highly effective for detecting improper tasks.

Analyzing the weights of each textual feature gives us to capture *red-flag* keywords that tend to appear in improper tasks; "account" and "password" appear frequently in the tasks asking for registration to particular web services, the term "e-mail" is common in the tasks collecting personal information, and "blog" and "open" are terms often found in tasks requesting the creation of blog accounts (Figure 2). In contrast, terms like "characters," "over," and "review" are repeatedly shown in proper tasks that often ask workers to "write a review in over N characters."

Figure 3, 4 and 5 show correctly classified and misclassified tasks. Typical improper tasks requiring account information for external web service, such as in Figure 3, were likely to be classified correctly. An example of improper task shown in Figure 4 asks workers to post a review on an external online review webpage; however, this task is misclassified because the task contains *good* terms such as "review," "characters," and "over," which frequently occur in proper tasks. A converse example is shown in Figure 5. In this proper task, workers are asked to give sample messages for opening a blog and the task contains some red-flag keywords such as "blog" and "open."



Additional offer! Check and hit 'Like!' button

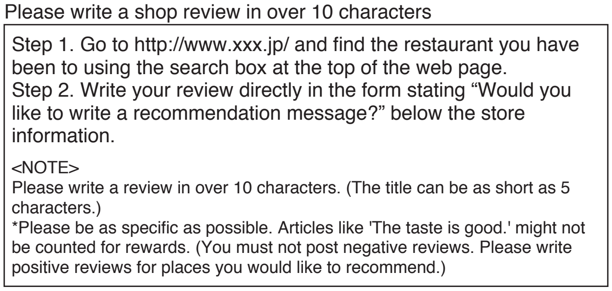
[1] Please search the keyword 'Unique Gifts Shopping' in Yahoo! or Google.
[2] Please check the name of the website (name of the shop).
[3] Please hit 'Like!'(on Facebook) at the bottom of the page.

Please fill in the name of the website in [2] below as well as your Facebook account.
* DO NOT create a new Facebook account for this task. This procedure is limited to people who currently have a Facebook account.

Name of website

Facebook account

Figure 3: Example of an improper task *correctly* classified as improper



Please write a shop review in over 10 characters

Step 1. Go to http://www.xxx.jp/ and find the restaurant you have been to using the search box at the top of the web page.
Step 2. Write your review directly in the form stating "Would you like to write a recommendation message?" below the store information.

<NOTE>
Please write a review in over 10 characters. (The title can be as short as 5 characters.)
*Please be as specific as possible. Articles like 'The taste is good.' might not be counted for rewards. (You must not post negative reviews. Please write positive reviews for places you would like to recommend.)

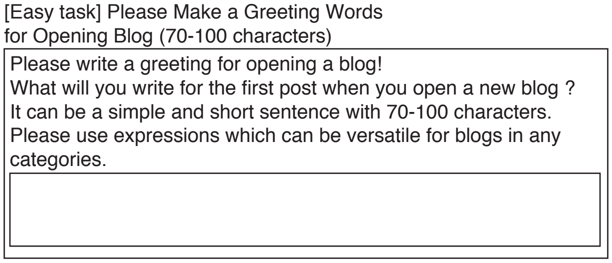
Figure 4: Example of an improper task *wrongly* classified as proper

We observed several characters of improper and proper tasks as well. Tasks that pay substantially higher rewards for workers are prone to be improper; proper tasks are more likely to hire trustworthy workers and filter workers considering historical approval rate or the verification status of the identification of workers. Further, requesters having high reputations usually post proper tasks.

4 Utilizing Non-expert Judgments for Training Classifiers

We investigate the idea of replacing the (expensive) expert labels with (cheap) non-expert labels collected in crowdsourcing marketplaces. Since the reliability of non-expert judgments depends on individual workers, we introduce re-

[4] http://www.csie.ntu.edu.tw/~cjlin/liblinear/



[Easy task] Please Make a Greeting Words
for Opening Blog (70-100 characters)

Please write a greeting for opening a blog!
What will you write for the first post when you open a new blog ?
It can be a simple and short sentence with 70-100 characters.
Please use expressions which can be versatile for blogs in any categories.

Figure 5: Example of a proper task *wrongly* classified as improper

Table 2: Performance comparison of the classifier trained only with expert judgments and the one trained only with non-expert judgments. The performance is evaluated in averaged AUC and its standard deviation.

Expert	Non-expert judgments only		
judgments only	Majority voting	Dawid and Skene	No aggregation
0.950 (±0.015)	0.759 (±0.052)	0.817 (±0.064)	0.754 (±0.051)

dundancy, that is, we assign each task to multiple crowdsourcing workers and aggregate their answers to obtain a more reliable label. We train a classifier using the obtained labels, and compare its detection performance with that of the model trained with the expert labels.

4.1 Aggregation of Crowdsourced Judgments

Since worker labels are not always as reliable as expert labels, we collect multiple labels $\{y_{i,j}\}_{j \in \mathcal{J}_i}$ from several workers for each task i and aggregate them to obtain more reliable labels $\{y_i\}_{i \in \{1,2,\cdots,N\}}$. The simplest aggregation strategy is to take majority voting; however, the abilities of workers vary significantly, as shown in Figure 6. To cope with such variation depending on individual workers, several sophisticated statistical methods considering worker abilities have been proposed (Dawid and Skene 1979; Whitehill et al. 2009; Welinder et al. 2010). In this study, we test two aggregation strategies—majority voting and the method of Dawid and Skene (1979), and a no-aggregation strategy, each of which is described below.

- **Majority voting**
  Aggregated label y_i is obtained by taking the majority in worker labels $\{y_{i,j}\}_{j \in \mathcal{J}_i}$, and is used as a class label in the training dataset $\{(x_i, y_i)\}_{i \in \{1,2,\cdots,N\}}$. Ties are resolved randomly.
- **Dawid and Skene (1979)**
  The approach of Dawid and Skene (1979), one of the popular methods for worker quality controls, models the ability of a worker with two parameters: the probability of the worker answering correctly when the true label is 1 and that when the true label is 0. The model parameters and the true labels are estimated with the EM algorithm, and the estimated true labels are used as the class labels in the training dataset.
- **No aggregation**
  This simple strategy adopted by Sheng, Provost, and Ipeirotis (2008) uses all of the given labels for the training dataset, which looks like $\{(x_i, y_{i,j})\}_{i \in \{1,2,\cdots,N\}, j \in \mathcal{J}_i}$.

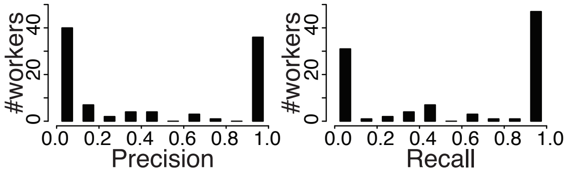


Figure 6: Histograms of precision and recall of worker ability evaluated with the expert labels. High variances in worker ability can be observed.

4.2 Results

We performed experiments following the same procedure as in Section 3.2, and the results are summarized in Table 2. The Dawid and Skene approach achieved the highest averaged AUC (0.817), followed by the majority voting approach. This is because the quality control methods compensated for the variances in the worker abilities shown in Figure 6, whereas the no-aggregation approach considers all workers as equal and suffers from the presence of low-quality workers.

5 Utilizing Expert and Non-expert Judgments for Training Classifiers

In the previous section, we investigated the approach of replacing the expert judgments with non-expert judgments. In this section, we proceed to combine both judgments to create a more accurate classifier. We describe candidate strategies for aggregating the expert and non-expert judgments and present the performance of the trained classifier.

5.1 Aggregation of Expert and Non-expert Judgments

Let us begin by considering the candidate approaches of combining the expert and non-expert judgments. For each task i, we have the expert label $y_{i,0}$ and worker labels $\{y_{i,j}\}$, or y_i if we aggregate the non-expert labels. For simplicity, let us denote the expert label as e and worker label(s) as $w = y_i$ (or $y_{i,j}$). The naïve methods to aggregate them perform either conjunction (AND) or disjunction (OR). These two methods produce different results when the expert and the non-expert disagree on the judgments (i.e., $e \neq w$). If we implement the conjunction strategy, we always choose 0 (proper) in the case of disagreement, and vice versa. Besides the conjunction and disjunction strategy, we could implement an approach that we skip samples for which the expert and the non-expert give the different judgments and do not include them to the training dataset.

We therefore have three strategies for the case of disagreement: select 0 as an agreed label (called N strategy), select 1 as an agreed label (P), skip the sample and do not include it in the training set (S). We could apply a different strategy for the case of $(e,w) = (0,1)$ and $(1,0)$, thus we have 9 strategies in total, that is, $\{\mathsf{N},\mathsf{P},\mathsf{S}\} \times \{\mathsf{N},\mathsf{P},\mathsf{S}\}$.

Next, we describe the detailed procedure of building a training set. Given an aggregation strategy of the expert and the non-expert judgments, we repeatedly process a task $i \in \{1, 2, \cdots, N\}$ as follows:

1. Set the expert label as $e = y_{i,0}$. If the worker labels are aggregated, set the worker label as $w = y_i$; otherwise process a worker label $w = y_{i,j}$ for each $j \in \mathcal{J}_i$.
2. If $e = w$, add $(e, \boldsymbol{x}_i)$ into the training data.
3. If $e \neq w$,
   - If strategy N is selected: add a sample $(0, \boldsymbol{x}_i)$ into the training set.
   - If strategy P is selected: add a sample $(1, \boldsymbol{x}_i)$ into the training set.
   - If strategy S is selected: skip the sample and do not add $(1, \boldsymbol{x}_i)$ into the training set.

Note that we could take the different strategies for the case of $(e,w) = (0,1)$ and $(1,0)$.

5.2 Results

We performed experiments following the same procedure as in Section 3.2. We applied the three non-expert label aggregation strategies described in Section 4.1 and the results are shown in Table 3. We can observe that the (S, P) strategy achieved the highest averaged AUC among all the strategies of non-expert judgments. This strategy always believes the label of *improper* given by the expert even if the non-expert disagrees, and skips the sample if the expert judges it as *proper* but the non-expert disagrees. The reasons behind the high performance of the (S, P) strategy might be explained as follows. We can always consider a task as improper if an expert judges it as such, irrespective of the non-expert label, thus, we can say that the expert has high *precision*. However, non-experts may have higher *recall* than experts, therefore we should consider the non-expert judgments of *improper* to create an accurate classifier. The performance of workers shown in Figure 6 supports this observation that, in fact, the workers have higher recall than precision on average.

We obtained the highest averaged AUC in the case of aggregating the non-expert labels by Dawid and Skene strategy and applying (S, P) strategy for the expert and non-expert aggregation (0.962). This classifier trained with the expert and non-expert judgments achieved a statistically significant improvement ($p < 0.05$ by Wilcoxon signed rank test) over the classifier trained with the expert judgments only.

Figure 7 shows the performance of classifiers with varying ratios of expert judgments used for training. This result shows that if the ratio of expert judgments is in 75%–100%, the accuracies of the classifiers are higher than the one trained with the expert judgments only (0.950, shown in Table 2). Thus, incorporating judgments by crowdsourcing non-expert workers allows for a reduction in the number of expert judgments by 25% while maintaining the level of detection performance.

Table 3: Performance comparison of the classifier trained with the expert and non-expert judgments with varying strategies for the expert and non-expert aggregation and the non-experts aggregation. N, P, and S denote the strategies of adopting 0 (proper) as an agreed judgment, adopting 1 (improper) as an agreed judgment, and skipping the sample, respectively. From 9 combinations, we omit (N, P) and (P, N) strategies because they are the same as the case of training with expert labels only and training with worker labels only. The performance is evaluated in averaged AUC and its standard deviation.

Aggregation strategy of expert and non-expert judgments		Aggregation strategy of non-expert judgments		
(e,w)		Majority	Dawid	No
$(0,1)$	$(1,0)$	voting	and Skene	aggregation
N	N	0.786 (±0.087)	0.763 (±0.076)	0.895 (±0.042)
N	S	0.816 (±0.081)	0.791 (±0.070)	0.929 (±0.029)
P	P	0.936 (±0.021)	0.951 (±0.017)	0.891 (±0.034)
P	S	0.790 (±0.047)	0.841 (±0.061)	0.829 (±0.046)
S	N	0.825 (±0.080)	0.816 (±0.075)	0.900 (±0.041)
S	P	**0.959** (±0.013)	**0.962** (±0.013)	**0.950** (±0.016)
S	S	0.877 (±0.033)	0.867 (±0.032)	0.935 (±0.026)

6 Related Work

One of the fundamental challenges in crowdsourcing is controlling the quality of the obtained data. Promising approaches for quality control can be categorized into task design (Kittur, Chi, and Suh 2008), worker filtering, and inter-agreement metrics of multiple submissions. Common techniques for worker filtering are summarized in (Kazai 2011), such as restricting workers by their historical approval rate or geo location, introducing pre-task qualification test and measuring the reliability of workers' submissions by evaluating the agreement to the gold standard data. Another widely used approach is to obtain multiple submissions from different workers and aggregate them by applying majority voting (Snow et al. 2008) or other sophisticated approaches. Dawid and Skene addressed the problem for aggregating medical diagnoses by multiple doctors to make more accurate decisions (1979). Whitehill et al. explicitly modeled the difficulty of each task (2009), and Welinder et al. introduced the difficulty of each task for each worker (2010). All the existing work listed here tackled the problem of *worker* quality control, whereas our work is the first to address the issue of *task* quality control.

Several studies address the problem of supervised learning from multiple labels obtained from crowd work-

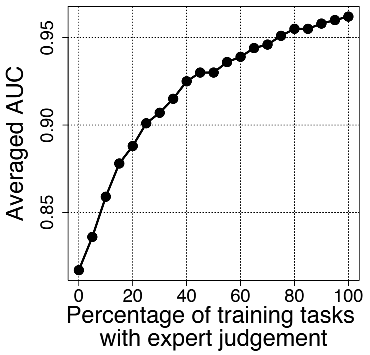


Figure 7: Averaged AUC of classifiers with varying number of training tasks with expert judgments. Dawid and Skene strategy for the non-expert labels aggregation and (S, P) strategy for the expert and non-expert aggregation are applied. The total number of tasks for training is $1,800$ among all the setting, and all the tasks are annotated by the workers. For example, if the ratio of tasks with expert judgment is 70%, the remaining 30% tasks have worker judgments only.

ers (Raykar et al. 2010; Yan et al. 2010; Kajino, Tsuboi, and Kashima 2012). Moreover, Tang and Lease (2011) and Kajino et al. (2012) focused on the problem of supervised learning with a setting where both expert and worker labels are available. This setting is similar to that in our work, and applying their methods might improve the performance of improper task detection.

7 Conclusion

We addressed the task quality control problem in crowdsourcing marketplaces, and presented our study on automatic detection of improper tasks. Our experimental results showed that the machine learning approach is highly effective in detecting improper tasks in a real crowdsourcing marketplace. We also investigated the approaches of leveraging crowdsourcing workers for task annotation, and applied quality control techniques to handle the variability of worker reliability. A classifier trained by both the expert and the non-expert workers achieved a statistically significant improvement. Our results also showed that the use of crowdsourced annotations allowed a reduction the number of expert judges by 25% while maintaining the level of detection performance.

We plan to extend our study to online monitoring, where tasks arrive sequentially and the annotations from operators and workers also arrive online. We will investigate effective online classification methods for improper task detection, and efficient usage of the classifier to reduce the monitoring costs of the operators in future work.

Acknowledgments

Y. Baba and H. Kashima were supported by the Funding Program for World-Leading Innovative R&D on Science and Technology (FIRST Program).

References

Dawid, A. P., and Skene, A. M. 1979. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statics)* 28(1):20–28.

Ipeirotis, P. G. 2010. Analyzing the amazon mechanical turk marketplace. *XRDS: Crossroads, The ACM Magazine for Students* 17(2).

Kajino, H.; Tsuboi, Y.; Sato, I.; and Kashima, H. 2012. Learning from Crowds and Experts. In *Proceedings of the 4th Human Computation Workshop (HCOMP).*

Kajino, H.; Tsuboi, Y.; and Kashima, H. 2012. A Convex Formulation for Learning from Crowds. In *Proceedings of the 26th Conference on Artificial Intelligence (AAAI).*

Kazai, G. 2011. In search of quality in crowdsourcing for search engine evaluation. *Advances in Information Retrieval* 165–176.

Kittur, A.; Chi, E.; and Suh, B. 2008. Crowdsourcing user studies with mechanical turk. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems (CHI).*

Raykar, V. C.; Yu, S.; Zhao, L. H.; Florin, C.; Bogoni, L.; and Moy, L. 2010. Learning From Crowds. *Journal of Machine Learning Research* 11:1297–1322.

Sheng, V. S.; Provost, F.; and Ipeirotis, P. G. 2008. Get Another Label? Improving Data Quality and Data Mining Using Multiple, Noisy Labelers. In *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD).*

Snow, R.; O'Connor, B.; Jurafsky, D.; and Ng, A. Y. 2008. Cheap and Fast – But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP).*

Sorokin, A., and Forsyth, D. 2008. Utility data annotation with Amazon Mechanical Turk. In *Proceedings of the 1st IEEE Workshop on Internet Vision.*

Tang, W., and Lease, M. 2011. Semi-Supervised Consensus Labeling for Crowdsourcing. In *ACM SIGIR Workshop on Crowdsourcing for Information Retrieval (CIR).*

Welinder, P.; Branson, S.; Belongie, S.; and Perona, P. 2010. The Multidimensional Wisdom of Crowds. In *Advances in Neural Information Processing Systems 23.*

Whitehill, J.; Ruvolo, P.; Wu, T.; Bergsma, J.; and Movellan, J. 2009. Whose Vote Should Count More: Optimal Integration of Labels from Labelers of Unknown Expertise. In *Advances in Neural Information Processing Systems 22.*

Yan, Y.; Rosales, R.; Fung, G.; Schmidt, M.; Hermosillo, G.; Bogoni, L.; Moy, L.; Dy, J.; and Malvern, P. 2010. Modeling Annotator Expertise: Learning When Everybody Knows a Bit of Something. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS).*