How to Assure the Quality of Human Computation Tasks When Majority Voting Fails?

Yu-An Sun
Xerox Innovation Group
800 Phillips Road
Webster, NY, USA
YuAn.Sun@xerox.com

Christopher R. Dance
Xerox Innovation Group
6 chemin de Maupertuis
38240 Meylan, France
Chris.Dance@xrce.xerox.com

Shourya Roy
Xerox Innovation Group
IIT Madras Research Park
Chennai, India
Shourya.Roy@xerox.com

Greg Little
MIT CSAIL
32 Vassar Street
Cambridge, MA, USA
glittle@gmail.com

Abstract

Quality assurance remains a key topic in human computation research. Prior work indicates that independent agreement is effective for low difficulty tasks, but performs poorly for moderately difficult tasks since the majority of responses may be inaccurate. We present experimental results showing that humans are better at identifying correct answers than producing correct answers in such moderately difficult tasks. This motivates us to propose a simple quality control process based on tournament selection. We demonstrate that tournament selection repeatably finds the correct answers in as few as 3 to 5 rounds. We also compare the efficiency of the tournament selection method and elimination selection method.

1 Introduction

Human computation is a growing research field. It holds the promise of humans and computers working seamlessly together to implement powerful systems, but it requires quality assurance to identify the correct results produced by the crowd. A common quality control method is redundancy (independent agreement) — let multiple people work on the same tasks and then take the majority answer as the correct result. However, this method does not address the common situation where most people produce the same incorrect answers, and only a small percentage of people produce better quality work. As observed in [10], the chance of obtaining independent agreement decreases as a question becomes more difficult or has too many possible correct answers.

This paper addresses this problem with a quality control process based on tournament selection. Tournament selection is commonly used with genetic algorithms to mimic survival of the fittest. By combining tournament selection with human computation tasks, the minority of the good quality work can prevail over a majority of incorrect work. In this paper, we focus on tasks involving translation, but the proposed method is generalizable to other difficult tasks which potentially have more than one correct answer. We also evaluate an alternative selection method known as elimination selection [8] to assess whether it can find the best option with a higher probability for a given number of comparisons.

Following a discussion of related work, we describe our quality control process, which is based on tournament selection. We then present results from human computation tasks, demonstrating that in situations where majority voting regularly fails, tournament selection and elimination selection repeatably find the correct answers in as few as 3 to 5 rounds. We conclude by discussing the cost of tournament selection and further research.

1.1 Related Work

Online labor marketplaces like Amazon Mechanical Turk allow companies or individuals posting jobs for a variety of tasks such as review collection[1], image labeling[2], user studies[3], word-sense disambiguation[4], machine translation evaluation[5] and EDA simulation[6], attracting millions of users from all over the world. While most users of online labor marketplaces have been successful in generating large volumes of data, quality assurance remains a central challenge. In a blog post, one author has compared AMT with a "market for lemons", following a seminal paper by the economist George Akerlof. In such a market, owing to lack of quality assurance and the absence of additional framework to filter out good input from junk, most requestors rely on redundancy followed by majority voting to ensure quality. In [7], it is reported that majority voting lead to best result in search result categorization task into four categories viz. off topic, spam, matching, and not-matching. Similar observations were reported in [4] for natural language evaluation tasks. However, the performance of voting algorithms is affected by collusion among participants. In [9], a novel algorithm is proposed to separate worker bias from errors with hidden Gold standard answers and used these errors in post-processing to estimate the true worker error-rate.

Most work on quality assurance for crowdsourcing has studied tasks with discrete answers and with one clear correct answer. Marketplaces like AMT are best suited for tasks in which there are bona fide answers, as otherwise users would be able to "game" the system and provide nonsense answers in order to decrease the time that they spend working and thus increase their rate of pay [3]. However, there are plenty of tasks such as translation which do not fall in this category, as variability in natural languages leads to multiple different but perfectly valid translations for a given sentence.

2 Tournament Selection and Results

2.1 Our Approach

Our approach is based on the hypothesis that humans are better at comparing results to pick the correct one than at producing the correct results themselves. It includes the following steps:

1. Start with one single human computation task.
2. Ask n people to do the task independently. One person can only work on one task in this set of tasks of size n. The variable n can be optimized for different types of human computation task.
3. Collect all results of total size n. Discard duplicate results, leaving only k unique answers ($k \leq n$).
4. *Tournament Selection*: Randomly pick two results from the k answers, ask one person to give a up-or-down vote on the comparison. The result that gets an up vote goes into a pool of "next generation" answers.
5. Repeat step 4 for n times, this generates a new pool of answers of size n.
6. Stopping Condition: Repeat steps 4-5 for m times. The stopping condition variable m may be optimized based on the type of human computation task.
7. After the stopping condition is met, a pool of final selected results is generated and the majority answer from this pool is identified as the final answer.

In all the experiments we described in this paper, variable n is set to be 30 and m is set to be 5.

2.2 Experimental Results

We ran the tournament selection algorithm of section 2.1 on tasks consisting of the translation of Chinese idioms. We started with the following distinct outputs generated by the crowd, with the fourth entry being the correct translation: Like a dog fails to draw a tiger, Who are you?, none, ***Attempting something beyond one's ability and fail***, painted tiger anti-dog, to try to draw a tiger and end up with the likeness of a dog. The initial sets of pairs were chosen by selecting uniformly at random from this pool and were presented to crowd who had to choose the better translation. Each chosen translation was put in a new pool and the set of pairs for the next round was chosen by selecting uniformly at random from this new pool. This process was repeated. At the end of each round, the proportion of each of the above entries was computed and the corresponding plot is shown in Figure 1. The correct translation was a minority to start off with but it gradually surpassed all other candidates and emerged as a clear winner within five rounds.

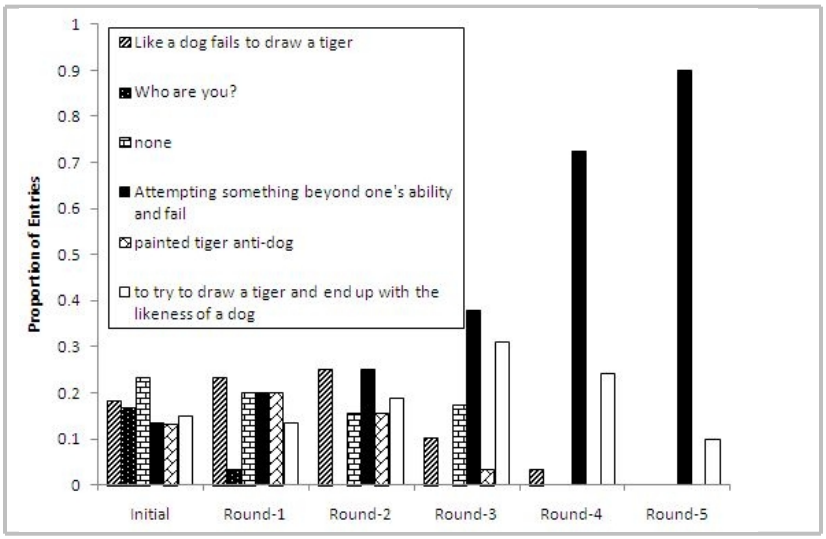

Figure 1: Result of tournament selection for one Chinese idiom.

3 Analysis of Results

3.1 Choice Probability Matrix

We analyze the selection between pairs of translated text. The following comparing matrix is calculated based on the experimental results. The number in each cell (*i, j*) represents the probability that the translation in row *i* is chosen as the correct one when the pair of translations *i, j* is compared by a crowd worker. We number the translations as follows: 1: none, 2: Who are you? 3: Painted tiger anti-dog, 4: Like a dog failed to draw a tiger, 5: to try to draw a tiger and end up with the likeness of a dog, and 6: Attempting something beyond one's ability and fail. For example, *(1, 2) = 0.75* represents the probability that "none" is chosen as a better translation over "Who are you?" 75% of the time.

Table 1: Pair Comparison Matrix

	1	**2**	**3**	**4**	**5**	**6**
1	-	0.75	0.45	0.36	0.14	0
2	0.25	-	0	0	Not drawn	0.5
3	0.55	1	-	0.91	0.2	0.5
4	0.64	1	0.09	-	0.11	0.08
5	0.86	Not drawn	0.8	0.89	-	0.3125
6	1	0.5	0.5	0.92	0.6875	-

3.2 Algorithm Testing: Elimination Selection Method

Using the above pairwise comparison matrix, we ran simulations to test a different selection method known as elimination selection [8]. In this method, pairs are selected at random from the set of candidate options, and any option that has lost *T* comparisons is eliminated as a candidate. The final answer is determined when there is only one translation left. Variable *T* can be optimized for different types of tasks.

Instead of conducting more crowdsourcing experiments, we used Table 1 to simulate the two algorithms. The simulation shows that with *T* set to be less than 16, the correct answer, ***Attempting something beyond one's ability and fail***, is not selected as the winner with high probability. Smaller *T* implies a smaller cost for the human computation task, but larger *T* represents a higher probability of picking the correct answer, assuming that the crowd actually does prefer the correct answer.

Table 2 summarizes the simulation results for different *T* averaged value over a total of 6 simulation runs. Comparing the result of elimination selection with tournament selection, we found that when *T = 18*, tournament selection requires 120 comparisons over 4 rounds to have

clear winner. Thus elimination selection results in a saving of 12%.

Table 2: Simulation Results

	T = 16	T = 17	T = 18	T = 19
Correctness	83.3%	83.3%	100%	100%
Total Number of Comparison Tasks	94	99.83	105.67	109.5

4 Conclusions and Further Work

From Figure 1, it is clear that after 3 rounds of tournament selection, the correct answer has emerged to be the winner. It only takes 5 rounds for the winner to reach 90% of population. Furthermore, Table 2 shows that an alternative selection method, elimination selection [8], can somewhat reduce the cost of quality control as measured in terms of the number of pairwise comparisons. Thus, we observe that translation is well-suited to the use of pairwise comparison methods for post-process quality control. This is because humans find it relatively easy to pick the right translation from a set, than to generate the right translation.

It would be of interest to explore methods for aggregating translations of whole paragraphs rather than short idioms. Furthermore, our vision is to be able to predict the outcome of an entire human computation algorithm from the properties of the components of this algorithm. This paper explored components that are based on majority voting and on pair-wise comparison. However, to achieve this vision the community will have to engage in substantial further experimental work to understand the behavior of other components such as answers to multiple choice questions and multi-way rankings.

References

[1] Su, Q., Pavlov, D., Chow, J., and Baker, W.C. 2007. Internet-scale collection of human-reviewed data. In Proc. Of the 16th International Conference on the World Wide Web (WWW-2007).

[2] Sorokin, A. and Forsyth, D. 2008. Utility data annotation with Amazon Mechanical Turk. In Proc. of the First IEEE Workshop on Internet Vision at CVPR-2008.

[3] Kittur, A., Chi, E. H., and Suh, B. 2008. Crowdsourcing user studies with Mechanical Turk. In Proc. of the 26th Annual ACM Conference on Human Factors in Computing Systems (CHI-2008).

[4] 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 Proc. Of EMNLP-2008.

[5] Callison-Burch, C. 2009. Fast, Cheap, and Creative: Evaluating Translation Quality Using Amazon's Mechanical Turk. In Proc. of EMNLP-2009.

[6] Andrew DeOrio and Valeria Bertacco. Human computing for EDA. In Proc. Of the 46th Annual Design Automation Conference (DAC-2009)

[7] Le, J., Edmonds, A., Hester, V., and Biewald, Lukas. Ensuring quality in crowdsourced search relevance evaluation. Workshop on Crowdsourcing for Search Evaluation. ACM SIGIR 2010

[8] M. Adler, P. Gemmell, M. Harchol-Balter, R. Karp, C. Kenyon. Selection in the Presence of Noise: The Design of Playoff Systems, Proceedings of the Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, 564–573, 1994.

[9] P. Ipeirotis, F. Provost, and J. Wang. Quality management on Amazon mechanical turk. In HCOMP, 2010

[10] G. Little and Y. Sun, Human OCR: Insights from a Complex Human Computation Process, Workshop on Crowdsourcing and Human Computation, Services, Studies and Platforms, ACM CHI, 2011.