Crowds, not Drones: Modeling Human Factors in Interactive Crowdsourcing

Senjuti Basu Roy[†], Ioanna Lykourentzou[††], Saravanan Thirumuruganathan[‡,△]
Sihem Amer-Yahia[◇], Gautam Das[‡,△].
[†]UW Tacoma, [††]CRP Henri Tudor/INRIA Nancy Grand-Est, [‡]UT Arlington, [△]QCRI, [◇] CNRS, LIG
senjutib@uw.edu, ioanna.lykourentzou@{tudor.lu,inria.fr},
saravanan.thirumuruganathan@mavs.uta.edu, sihem.amer-yahia@imag.fr,
gdas@uta.edu

ABSTRACT

In this vision paper, we propose `SmartCrowd`, an intelligent and adaptive crowdsourcing framework. Contrary to existing crowdsourcing systems, where the process of hiring workers (crowd), learning their skills, and evaluating the accuracy of tasks they perform are fragmented, siloed, and often ad-hoc, `SmartCrowd` foresees a paradigm shift in that process, considering unpredictability of human nature, namely *human factors*. `SmartCrowd` offers opportunities in making crowdsourcing intelligent through iterative interaction with the workers, and adaptively learning and improving the underlying processes. Both existing (majority of which do not require longer engagement from volatile and mostly non-recurrent workers) and next generation crowdsourcing applications (which require longer engagement from the crowd) stand to benefit from `SmartCrowd`. We outline the opportunities in `SmartCrowd`, and discuss the challenges and directions, that can potentially revolutionize the existing crowdsourcing landscape.

1. INTRODUCTION

Crowdsourcing systems have gained popularity in a variety of domains. Common crowdsourcing scenarios include data gathering (asking volunteers to tag a picture or a video), document editing (as in Wikipedia), opinion solicitation (asking foodies to provide a summary of their experience at a restaurant), collaborative intelligence (asking residents to match old city maps), etc. The action of each worker involved in crowdsourcing can be viewed as *an approximation of ground truths*. In the examples we describe, truth could be a complete set of tags describing a picture, a Wikipedia article, an exhaustive opinion on a restaurant, etc. Truth can be objective (single ground truth) or subjective, where there may be different truths for different users (e.g., youngsters tend to like fast-food restaurants while young professionals may not, photography professionals tend to prefer tags reflecting photo quality as opposed to photo content). In this paper, we are interested in the question of harnessing the crowd to approximate truth(s) effectively and efficiently while taking into account the innate uncertainty of human behavior, named human factors.

Crowdsourcing Today: Existing systems are built on top of private or public platforms, such as Mechanical Turk, Turkit, Mob4hire, uTest, Freelancer, eLance, oDesk, Guru, Topcoder, Trada, 99design, Innocentive, CloudCrowd, and CloudFlower [3]. Tasks are typically small, independent, homogeneous, have minor incentives, and do not require longer engagement from workers. Similarly, the crowd is typically volatile, arrival and departure is asynchronous, with different levels of attention and accuracy.

Limitations of current approaches: There are two primary limitations related to current crowdsourcing approaches. The first refers to the separation and non-optimization of the underlying processes in a dynamic environment. The second limitation is related to the omission of human factors when designing an optimized crowdsourcing solution. In fact, while recent research investigates some of the optimization aspects, those aspects are not studied in conjunction with human factors.

Three major processes involved in the task of ground-truth approximations are - worker skill estimation, worker-to-task assignment, and task accuracy evaluation. Most current commercial crowdsourcing systems (a survey of which can be found in [3]) either do not offer algorithmic optimization, or do that partially and in isolation. Pre-qualification tests, the usage of golden standard data, or hiring of workers based on worker past performance are the norm. Task assignment is completely open and allows self-appointment by the workers, thus undermining quality (workers prefer to increase their individual profit over accomplishing qualitative tasks). Worker wage is often pre-determined and fixed per task, oblivious to the quality of the actual pool of workers who undertake the task in reality. Recent research undertakes some of the challenges unsolved by commercial platforms, and proposes active learning strategies for task evaluation [10, 1, 7], task assignment process [5], adjusting worker wages accordingly to skills [11]. However these works: i) focus on a specific crowdsourcing application type (mostly real-time crowdsourcing with highly volatile crowds) thus losing genericity, and ii) focus on the algorithmic optimization of some but not all of the involved processes (e.g. skill learning, or wage determination, or task assignment).

A more critical limitation refers to the omission or inadequate incorporation of the uncertainty stemming from human factors into the design of the crowdsourcing optimization algorithm. Algorithmic solutions rely on simple, idealized models (e.g. known worker skills or steady worker performance). A recent work [8] proposes probabilistic worker skill estimation models, based on the workers past performance, considering potential deviations in worker performance. Another recent work studies the egoistic profit-oriented objectives of individual workers to incentivize them (e.g. by properly adjusting wages) in order to calibrate algorithms that approximate the ground truth related to the crowdsourcing task [2]. Benefit of explicit feedback and information exchange between workers is studied [4, 6] to improve worker self-coordination, but no existing research incorporates these aspects in a dynamic and interactive environment, nor are there optimized solutions for ground truth discovery, considering human factors.

Opportunities: Future crowdsourcing systems therefore need to, first treat the crowdsourcing problem not in optimization silos, but as an adaptive optimization problem, seamlessly handling the three main crowdsourcing processes (worker skill estimation, task assignment, task evaluation). Secondly and equally important, the uncertainty stemming from human factors needs to be quantified and incorporated into the design of any future algorithm that seeks to optimize the above adaptive crowdsourcing problem. For example, the estimation of every worker parameter that can be influenced by uncertainty needs to be incorporated into the design of the crowdsourcing optimization process. Also, the planning horizon and the optimization boundaries of any algorithm applied to facilitate crowdsourcing need consequently to be determined with this uncertainty in mind. New challenges rise from the above two opportunities, of adopting a seamless crowdsourcing process and of incorporating uncertainty into it.

In summary, crowdsourcing has transitioned from being used as research tool into a research topic on its own. Sooner or later, database researchers have to confront the issues resulting from hybrid processing involving humans and computers. The uncertainties arising due to human factors in crowdsourcing are very different from traditional uncertainty, such as in probabilistic databases [9]. `SmartCrowd` envisions crowdsouring as an adaptive process where human factors are given the significance they deserve. Further, we also introduce a mechanism of crowd-indexing by which workers are organized into groups. Such indices are triggered by human factors, dynamically maintained and provide an efficient way to search for workers.

2. OUR VISION

We propose to rethink crowdsourcing as an adaptive process that relies on an interactive dialogue between the workers and the system in order to build and refine worker skills, while tasks are being completed. In parallel, as workers complete more tasks, the system 'learns" their skills more accurately, and this adaptive learning is used to dynamically assign tasks to workers in the next iteration, by understanding the intrinsic uncertainty of human behavior. Note that, key to the success of these steps is the knowledge on ground truth, which the system is oblivious of (and wishes to discover) in the first place. The primary paradigm shift in `SmartCrowd` is in envisioning the process of ground-truth discovery to be dynamic, adaptive, and iterative in discovering skills required for tasks, evaluating the accuracy of completed tasks, learning skills of involved workers, assigning tasks to workers, determining the number of workers and offered incentives, considering human factors. Interestingly, these intermediate objectives are often inter-dependent, and improving one improves others. The overall objective of this adaptive process is *to maximize accuracy and efficiency while reducing cost and effort.*

2.1 High Level Architecture

The primary distinction of our framework is the deliberate acknowledgement of the importance of human factors in crowdsourcing and how it guides each of our objectives in a dynamic environment. Further, we envision our framework to have an interactive dialogue with the workers to enable adaptive learning, while the workers participate in crowdsourcing tasks. The first two dimensions we tackle are:

- **"who knows what"**, i.e. to evaluate the contributions of workers and based on that to estimate their skills with the least possible error (skill learning process).
- **"who will be asked to contribute to what"**, i.e., by learning required skills for tasks and estimating workers' skills, assign tasks to workers (task assignment process).

`SmartCrowd` functions as follows: workers enter the crowdsourcing platform and complete tasks. Many crowdsourced tasks typically require multiple skills. In the beginning, `SmartCrowd` holds no knowledge over the skills of newcomers. Furthermore, some required skills may be latent, and unknown to `SmartCrowd` in the beginning. As the workers undertake and complete more tasks, `SmartCrowd` discovers latent skills, evaluates workers contribution to the tasks and learns their skills, and therefore assign appropriate tasks to the workers, which in turn achieves higher accuracy and improved efficiency in the process. Moreover, this process is adaptive and iterative, worker skills are "learnt more accurately" and "used more appropriately" over time, ensuring gradual improvement.

Figure 1 shows two primary functionalities that are improved adaptively in `SmartCrowd`: one depicting learning worker skills, and the other depicting completion time of the (ground truth discovery) tasks. More precisely, the steeper the skill estimation error curve gets, the faster we arrive to accurate approximation of workers' skills, i.e., the faster we can profile workers with low error. Also, there is a moment in time when the approximation error in skill estimation is acceptable. This is marked in the figure with a dashed vertical line. Before that, the system is in "cold start" phase, and does not know "much" about workers. Traditionally, this problem is tackled with uniform-prior assumptions, spammer-hammer model, multi-dimensional wisdom of crowd to bootstrap user skills [3]. After that, the framework continues to improve its knowledge on workers' skills and adaptively assigns tasks to workers in iteration, until the system determines that a stopping condition has been reached. Interestingly, faster minimization of skill estimation error leads to earlier termination of cold start period (i.e., the dashed vertical line to the left), which gives rise to better opportunities in designing the task assignment process (task assignment improvement area).

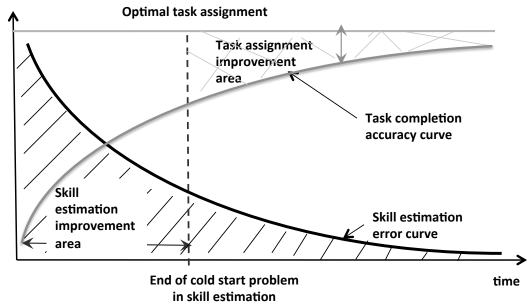

Figure 1: Tradeoff between Skill Estimation Accuracy and Task Completion Efficiency

As skill estimation improves, task completion efficiency is also expected to improve, since the system can assign tasks more intelligently to workers. However, worker skill estimation is critically related to accurate *task evaluation process*, i.e., to evaluate the accuracy of the completed tasks by the workers. In the absence of explicit ground truth, `SmartCrowd` resorts to uncovering the ground truth using workers themselves. While this interactive process does not necessarily require longer engagement from the workers in the system, it offers opportunities for improved learning. Therefore, the third and final dimension we tackle is:

- **"engaging workers explicitly to improve learning"**, i.e., how to further exploit the learned expertise of workers by engaging them explicitly in evaluating the skill of other workers or by completing more tasks.

Most importantly, these dimensions in `SmartCrowd` are studied in conjunction with two key aspects that are exclusive to crowdsourcing - **human factor** and **scale**. The unpredictability and inconsistency in human behavior are deliberate in the design of `SmartCrowd`. Additionally, `SmartCrowd` envisions the designed solutions to be scalable, i.e., tolerant to the size of the crowd, and its volatility. To the best of our knowledge, `SmartCrowd` is the first ever framework that considers these factors explicitly in crowdsourcing. Finally, `SmartCrowd` could be adapted inside existing systems, since it is designed assuming current crowdsourcing infrastructure.

In summary, to design accurate and efficient crowdsourcing, `SmartCrowd` relies on a formal modeling of the **task evaluation**, **worker skill estimation**, and **task assignment** processes, considering **human factor** and **scale**.

3. CHALLENGES AND DIRECTIONS

While the opportunities foreseen in `SmartCrowd` are novel, the challenges in achieving them are exceptionally arduous. These challenges get further magnified, because of, (1) *Human factor* - which necessitates the key challenges to be modeled and solved considering unpredictability and inconsistency in worker behavior, their volatility, and asynchronous arrival and departure; (2) *Scale* - which necessitates the solutions to be incremental and tolerant to the volatility of the crowd and its size. `SmartCrowd` proposes novel indexing opportunities and reasons that human factor induced crowd-indexing provides a transparent way of achieving the objectives of `SmartCrowd` in conjunction with human factors and scale.

3.1 Human Factors

Human factors, a key distinction of `SmartCrowd`, relates to the uncertainty and non-deterministic nature of the behavior of human workers. For example, there is uncertainty regarding worker availability: workers can enter the crowdsourcing platform when they want, remain connected for as long as they like and they may or may not accept to make a contribution. In the same sense, there is uncertainty regarding the wage that workers may request: worker wage may vary from person to person, even among persons with the same profile for the system, but also wage may vary for the same person in different times, for example due to the person's workload, available time but also due to unseen factors. Finally, uncertainty also goes for skills: the efficiency with which a person completes a task cannot be considered fixed and it is rather uncertain, for example it may decline with the previous workload of the person, or it may depend on the offered wage or on the worker's motivation and personal engagement in the task.

The uncertainty stemming from the human factors does not preclude from designing a crowdsourcing solution with a global optimization target. What it does mean, however, is that, instead of fixed parameter values, `SmartCrowd` needs to study the aforementioned dimensions considering probabilities and confidence boundaries (e.g. we cannot determine the "exact wage" of a person but an approximation, with certain deviation of a central wage value), and be able to update the probabilities, as workers complete more tasks.

3.2 Who Evaluates What and How?

Tasks submitted by workers need to be evaluated for accuracy. Interestingly, the process of evaluating completed tasks is tightly coupled with acquiring each worker's contribution, which in turn helps learning worker skills. A question however is, who evaluates what and how?

A worker's contribution to a task can be evaluated through a *fully-automated and implicit* way by comparing submitted results against each other. In lieu of a known ground truth, a worker's contribution could be measured by computing the divergence of submitted contributions thus far using simple or weighted averages, majority voting, etc. More sophisticated models such as multivariate data analysis could also be used to approximate ground truth. In all cases, implicit evaluation becomes effective when the acquired aggregated data approximates the unknown ground truth. A faster, more reliable but costlier alternative is to *explicitly* designate some of the current workers as the evaluators of submitted tasks.

We envision a *hybrid* method instead; task evaluation is performed by combining system's acquired intelligence augmented with explicit human expertise. This requires complex modeling - 1) how to combine implicit and explicit evaluations together, 2) when and how to hire explicit evaluators, 3) how many explicit evaluators are required. In addition, human factors also contributes multiple new parameters such as 4) what should be the offered incentives, 5) how to model inconsistent attention and arbitrary departure of explicit evaluators, and 6) how to compute this incrementally, as workers enter and exit asynchronously.

3.3 How to Estimate Worker Skills?

Skill estimation pertains to *learning* worker skills accurately and effectively. In `SmartCrowd`, the output of task evaluation (i.e., a worker's contribution to each completed task) is used to estimate worker skills. Therefore, the first challenge is, how to identify and quantify a skill set?

For many complex tasks, some skills may be *latent*. For example, in image moderation, skills might vary for different images. In SmartCrowd, we envision learning such latent skills as the tasks are being executed by workers. Discovering a set of latent skills could be formulated as a structure learning problem in machine learning with the objective of uncovering a multi-layer probabilistic model. On the contrary, the problem could also be formulated as a fixed probabilistic model with the objective of learning inference from it. Unlike traditional machine learning problems where the end objective is accurate prediction, one unique requirement for SmartCrowd is to make these discovered skills *contextual* and interpretable by the applications.

Irrespective of the specific algorithm used to quantify worker skills, additional challenges in the model involve - 1) determining the minimal number of tasks that workers (or certain groups of workers) need to complete, until their skills can be estimated with high accuracy, considering they may not behave consistently, 2) identifying the "stopping condition" to decide whether a worker's skills have been estimated with adequate certainty or not, and 3) enabling fast and incremental computation (using worker clustering or view maintenance) of skills, as new workers arrive. In addition, human factors causes additional challenges such as identifying declination of skills (possibly due to boredom) or model how worker skill changes over time.

3.4 How to Assign Tasks to Workers?

In SmartCrowd, we envision that workers are assigned to tasks based on learned workers' skills and the remaining unfinished tasks. Interestingly, unlike traditional task assignment problems in project management, in SmartCrowd , workers' skills are unknown in the beginning, and learned skills evolve as workers engage in more tasks and subject to inconsistency and unpredictability due to human factors.

In SmartCrowd, we model assigning tasks to workers as a *probabilistic optimization problem*, with the objective of maximizing accuracy, or minimizing time, or optimizing both at the same time probabilistically. Furthermore, additional factors such as cost (money) could be considered.

Several related questions (or constraints) are required to be factored into this formulation as well - (1) what if a worker declines an assigned task, 2) can multiple tasks be allocated to the same worker, 3) in the case of multiple task allocation, does SmartCrowd suggest an ordering tasks to the worker, 4) during task assignment, does SmartCrowd need to assign tasks such that there are no idle workers, 5) is there an upper limit on the number of tasks that a single worker can be assigned to in one iteration? 6) how important is the system's benefit vs worker's benefit? Should the system optimize across tasks (i.e., exploit), or give newcomers opportunities (i.e, explore) to prove their skills?

3.5 Crowd-Indexing

Crowdsourcing is an adaptive process - where workers/tasks arrive asynchronously, and the system learns more about workers as they complete assigned tasks. Satisfying the key objectives of worker skill estimation, worker-to-task assignment, and task accuracy evaluation while accounting for human factors at scale, necessitates the development of efficient searching techniques. SmartCrowd proposes crowd-indexing to that end, where workers are organized and indexed into *groups*, and the indexes are dynamically maintained.

Interestingly, SmartCrowd demands new forms of indexing triggered by human factors, such as predictive skill estimation and task acceptance rate. These factors are dynamic and vary over time, as workers undertake and complete more tasks. Efficient determination of the right group of workers for collaborative tasks is a key question when optimizing cost (time and money). Similarly, selecting explicit evaluator(s) efficiently for task evaluation could benefit tremendously from index design. However, in SmartCrowd, we envision incremental indexing strategies, that are adaptive to this dynamic environment.

In contrast to traditional database indexing, crowd-indexing is (a) on-demand indexing where the notion of query workload is akin to tasks arriving at different rates (b) constrained indexing with different objectives such as latency, budget, worker skill diversity (c) alternate indexing as it requires to have a fall-back option (due to the uncertainty of workers accepting a task).

4. CONCLUSION

In this paper, we developed a vision for intelligent crowdsourcing and presented our framework, SmartCrowd. In contrast to existing systems, SmartCrowd promotes an iterative interaction with workers and an involvement of those workers beyond task completion (they are involved in evaluating each others' contributions), in order to adaptively learn and improve the processes of learning workers' skills and assigning tasks. Both existing (which do not require longer engagement from a volatile and mostly non-recurrent workers) and next generation crowdsourcing applications (which require longer engagement from the crowd) could benefit from our vision. As discussed in this paper, increasing intelligence in SmartCrowd comes with several hard challenges. SmartCrowd aims to be principled yet efficient in proposing the solution to those challenges.

References

[1] R. Boim, O. Greenshpan, T. Milo, S. Novgorodov, N. Polyzotis, and W. C. Tan. Asking the right questions in crowd data sourcing. In *ICDE*, pages 1261–1264, 2012.

[2] R. Cavallo and S. Jain. Efficient crowdsourcing contests. In *AAMAS*, pages 677–686, Richland, SC, 2012.

[3] A. Doan, R. Ramakrishnan, and A. Y. Halevy. Crowdsourcing systems on the World-Wide Web. *Communications of The ACM*, pages 385–396, 2011.

[4] S. Dow, A. Kulkarni, S. Klemmer, and B. Hartmann. Shepherding the crowd yields better work. In *CSCW*, 2012.

[5] C.-J. Ho and J. W. Vaughan. Online task assignment in crowdsourcing markets. In *AAAI*, 2012.

[6] S.-W. Huang and W.-T. Fu. Don't hide in the crowd!: increasing social transparency between peer workers improves crowdsourcing outcomes. In *CHI*, 2013.

[7] D. R. Karger, S. Oh, and D. Shah. Budget-optimal task allocation for reliable crowdsourcing systems. *CoRR*, abs/1110.3564, 2011.

[8] X. Liu, M. Lu, B. C. Ooi, Y. Shen, S. Wu, and M. Zhang. Cdas: a crowdsourcing data analytics system. *Proc. VLDB Endow.*, 5(10):1040–1051, June 2012.

[9] A. Parameswaran and N. Polyzotis. Answering queries using humans, algorithms and databases. In *CIDR 2011.*

[10] A. Ramesh, A. Parameswaran, H. Garcia-Molina, and N. Polyzotis. Identifying reliable workers swiftly. Technical report, Stanford University.

[11] Y. Singer and M. Mittal. Pricing mechanisms for crowdsourcing markets. WWW '13, pages 1157–1166, 2013.