Quizz: Targeted Crowdsourcing with a Billion (Potential) Users

Panagiotis G. Ipeirotis*
Google & New York University
panos@stern.nyu.edu

Evgeniy Gabrilovich
Google
gabr@google.com

ABSTRACT

We describe Quizz, a gamified crowdsourcing system that simultaneously assesses the knowledge of users and acquires new knowledge from them. Quizz operates by asking users to complete short quizzes on specific topics; as a user answers the quiz questions, Quizz estimates the user's competence. To acquire new knowledge, Quizz also incorporates questions for which we do not have a known answer; the answers given by competent users provide useful signals for selecting the correct answers for these questions. Quizz actively tries to identify knowledgeable users on the Internet by running advertising campaigns, effectively leveraging the targeting capabilities of existing, publicly available, ad placement services. Quizz quantifies the contributions of the users using information theory and sends feedback to the advertising system about each user. The feedback allows the ad targeting mechanism to further optimize ad placement.

Our experiments, which involve over ten thousand users, confirm that we can crowdsource knowledge curation for niche and specialized topics, as the advertising network can automatically identify users with the desired expertise and interest in the given topic. We present controlled experiments that examine the effect of various incentive mechanisms, highlighting the need for having short-term rewards as goals, which incentivize the users to contribute. Finally, our cost-quality analysis indicates that the cost of our approach is below that of hiring workers through paid-crowdsourcing platforms, while offering the additional advantage of giving access to billions of potential users all over the planet, and being able to reach users with specialized expertise that is not typically available through existing labor marketplaces.

1. INTRODUCTION

Crowdsourcing has been the primary enabling mechanism behind the generation of many valuable Internet resources. Wikipedia, Freebase, and other knowledge repositories were created by volunteers who contributed knowledge about a wide variety of topics. Other human computation applications engage users in creative ways to generate interesting and useful by-products of the engagement. The ESP Game [33] asks users to participate in a game that generates useful image tags. ReCAPTCHA [34] verifies that users are humans by asking them to transcribe letters from a distorted image, thus helping with the digitization of books. Duolingo[1] teaches users a new language and as a by-product generates translations of written material in different languages. However, despite these widely-known success stories, building and engaging a community of users is a challenging task. Coming up with creative engagement strategies (e.g., ESP Game) is difficult, and spawning successful crowd-powered sites such as Wikipedia often seems like the exception rather than the norm.

In order to sidestep the problem, many efforts rely on paid crowdsourcing; for example, hiring workers through platforms such as Amazon Mechanical Turk allows for direct engagement of users, with a clear monetary incentive. Unfortunately, the introduction of money as a predictable and repeatable motivator is a mixed blessing. Users who are motivated by monetary rewards are often different than the users who are unpaid and motivated by other means [22, 28, 40]. Furthermore, studies indicate that *the use of monetary rewards can be highly detrimental for users who are already intrinsically motivated* [4]: the introduction of monetary compensation reveals to the users exactly how much their work is valued by the task requester, and low payments make things worse [16].

Finally, even if the incentive problem is solved, how does one attract the crowd that is properly qualified for a given task? The workers participating in paid crowdsourcing are typically non-expert users, and often lack the skills needed for the crowdsourcing effort. For example, if one's task calls for Swahili speakers or maxillofacial surgeons, then most labor marketplaces do not even provide access to such users, or only have few users with required expertise.

Thus, a set of natural challenging questions emerge. Can we replicate the predictability of paid crowdsourcing in terms of attracting participation, while engaging *unpaid* users? And how can we identify and incentivize *experts* among these users, who match the needs of the application at hand?

Here we propose to use existing Internet advertising platforms for targeting and attracting users, *with the suitable expertise for the task at hand*. Over the last decade, advertising platforms have improved their targeting capabilities to identify users who are good matches for the goals of the

*Work done while visiting Google.

[1] http://www.duolingo.com/

WWW'14, April 7–11, 2014, Seoul, Korea.
ACM 978-1-4503-2744-2/14/04.
http://dx.doi.org/10.1145/2566486.2567988 .

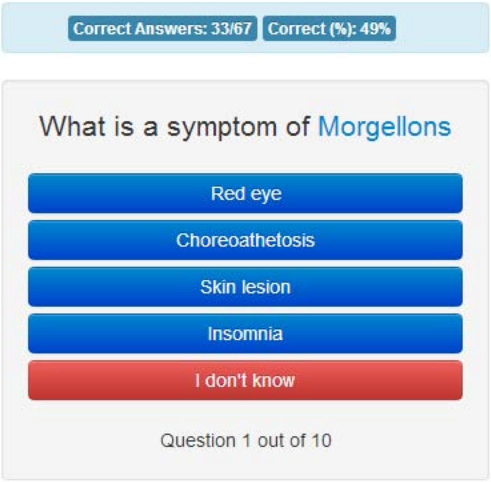


Figure 1: Screenshot of the Quizz system.

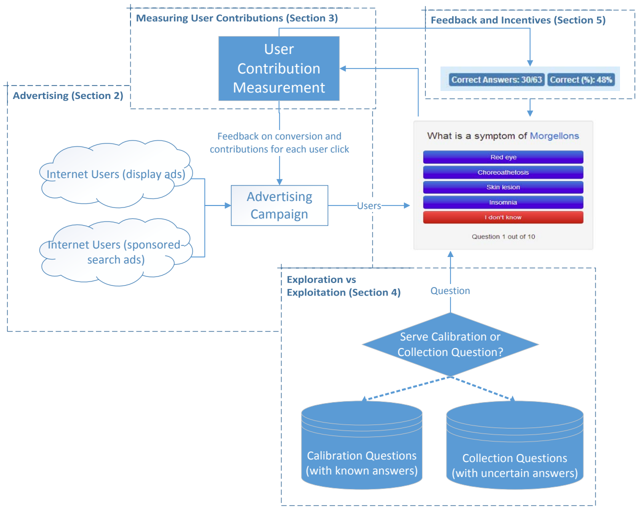

Measuring User Contributions (Section 3)
User Contribution Measurement
Feedback and Incentives (Section 5)
Correct Answers: 30/63
Correct (%): 48%
Advertising (Section 2)
Feedback on conversion and contributions for each user click
Internet Users (display ads)
Internet Users (sponsored-search ads)
Advertising Campaign
Users
What is a symptom of Morgellons
Red eye
Choreoathetosis
Skin lesion
Insomnia
I don't know
Question 1 out of 10
Exploration vs Exploitation (Section 4)
Question
Serve Calibration or Collection Question?
Calibration Questions (with known answers)
Collection Questions (with uncertain answers)

Figure 2: An overview of the Quizz system.

advertiser. In our case, we initiate the process with simple advertising campaigns but also integrate the ad campaign with the crowdsourcing application, and provide *feedback* to the advertising system for each ad click: The feedback indicates whether the user, who clicked on the ad, "converted" and the total contributions of the crowdsourcing effort. This allows the advertising platform to naturally identify websites with user communities that are good matches for the given task. For example, in our experiments with acquiring medical knowledge, we initially believed that "regular" Internet users would not have the necessary expertise. However, the advertising system automatically identified sites such as Mayo Clinic and HealthLine, which are frequented by knowledgeable consumers of health information who ended up contributing significant amounts of high-quality medical knowledge. Our idea is inspired by Hoffman et al. [17], who used advertising to attract users to a Wikipedia-editing experiment, although they did not attempt to target users nor attempted to optimize the ad campaign by providing feedback to the advertising platform.

Once users arrive at our site, we need to engage them to contribute useful information. Our crowdsourcing platform, *Quizz*, invites users to test their knowledge in a variety of domains and see how they fare against other users. Figure 1 shows an example question. Our quizzes include two kinds of questions: *Calibration* questions have known answers, and are used to assess the expertise and reliability of the users. On the other hand, *collection* questions have no known answers and actually serve to collect new information, and our platform identifies the correct answers based on the answers provided by the (competent) participants. To optimize how often to test the user, and how often to present a question with an unknown answer, we use a Markov Decision Process [29], which formalizes the exploration/exploitation framework and selects the optimal strategy at each point.

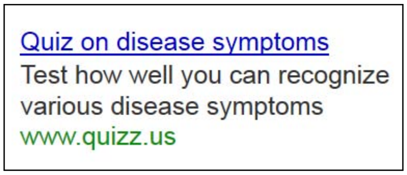
As our analysis shows, a key component for the success of the crowdsourcing effort is not just getting users to participate, but also to keep the good users participating for long, while gently discouraging low-quality users from participating. In a series of controlled experiments, involving tens of thousands of users, we show that a key advantage of attracting unpaid users through advertising is the strong self-selection of high-quality users to continue contributing, while low-quality users self-select to drop out. Furthermore, our experimental comparison with paid crowdsourcing (both paid hourly and paid piecemeal) shows that our approach dominates paid crowdsourcing both in terms of the quality of users *and* in terms of the total monetary cost required to complete the task.

The contributions of this paper are fourfold. First, we formulate the notion of *targeted crowdsourcing*, which allows one to identify crowds of users with desired expertise. We then describe a practical approach to find such users at scale by leveraging existing advertising systems. Second, we show how to optimally ask questions to the users, to leverage their knowledge. Third, we evaluate the utility of a host of different engagement mechanisms, which incentivize users to contribute more high-quality answers via the introduction of short-term goals and rewards. Finally, our empirical results confirm that the proposed approach allows to collect and curate knowledge with accuracy that is superior to that of paid crowdsourcing mechanisms at the same or lower cost.

Figure 2 shows the overview of the system, and the various components that we discuss in the paper. Section 2 describes the use of advertising to target promising users, and how we set up the campaigns to allow for continuous, automatic optimization of the results over time. Section 3 shows the details of our information-theoretic scheme for measuring the expertise of the participants, while Section 4 gives the details of our exploration-exploitation scheme. Section 5 discusses our experiments on how to keep users engaged, and Section 6 gives the details of our experimental results. Finally, Section 7 describes related work, while Section 8 concludes.

2. ADVERTISING FOR TARGETING USERS

A key problem of every crowdsourcing effort is soliciting users to participate. At a fundamental level, it is always preferable to attract users that have an inherent motivation for participation. Unfortunately, repeating the successes of efforts such as Wikipedia, TripAdvisor, and Yelp seems more of an art than a science, and we do not yet fully understand



Quiz on disease symptoms
Test how well you can recognize
various disease symptoms
www.quizz.us

Figure 3: Example ad to attract users

how to create engaging and viral crowdsourcing applications in a replicable manner. The emergence of paid crowdsourcing (e.g., Amazon Mechanical Turk) allows direct engagement of users in exchange for monetary rewards. However, the population of users who participate due to extrinsic rewards is typically different from the users who participate because of their intrinsic motivation.

Quizz uses online advertising to attract *unpaid* users to contribute. By running ads, we get into the middle ground between paid and unpaid crowdsourcing. Users who arrive at our site through an ad are not getting paid, and if they choose to participate they obviously do so because of their intrinsic motivation. This removes some of the wrong incentives and tends to alleviate concerns about indifferent users that "spam" the results just to get paid, or about workers that are trying to do the minimum work necessary in order to get paid. Thanks to the sheer reach of modern advertising platforms, the population of unpaid users can potentially be orders of magnitude larger than that in paid marketplaces. There are billions of users reachable through advertising, while even the biggest crowdsourcing platforms have at most a million users, many of them inactive [19, 18]. Therefore, if the need arises (and subject to budgetary constraints), our approach can elastically scale up to reach almost arbitrarily large populations of users, by simply increasing the budget allocated to the advertising campaign. At the same time, we show in Section 6 that our approach allows efficient use of the advertising budget (which is our only expenditure), and our overall costs are the same or lower than those in paid crowdsourcing installations.

A significant additional benefit of using an advertising system is its ability to *target* users with expertise in specific topics. For example, if we are looking for users possessing medical knowledge, we can run a simple ad like the one in Figure 3. To do so, we select keywords that describe the topic of interest and ask the advertising platform to place the ad in relevant contexts. In this study, we used Google AdWords[2], and opted into *both search and display ads*, while in principle we can use any other publicly available advertising system.

Selecting appropriate keywords for an ad campaign is a challenging topic in itself [13, 1, 20]. However, we believe that trying to optimize the campaign only through manually fine-tuning its keywords is of limited utility. Instead, we propose to automatically optimize the campaign by quantifying the behavior of the users that clicked on the ad. A user who clicks on the ad but does not participate in the crowdsourcing application is effectively "wasting" our advertising budget; using the advertising terminology, such user has not "converted." Since we are not just interested in attracting any users but are interested in attracting users who contribute, we use Google Analytics[3] to track user conversions. Every time a user clicks on the ad and then participates in a quiz, we record a conversion event, and send this signal back to the advertising system. This way, we are effectively asking the system to optimize the advertising campaign for maximizing the number of conversions and thus increasing our contribution yield, instead of the default optimization for the number of clicks.

Although optimizing for conversions is useful, it is even better to attract *competent* users (as opposed to, say, users who just go through the quiz without being knowledgeable about the topic). That is, we want to identify users who are both willing to participate *and* possess the relevant knowledge. In order to give this refined type of feedback to the advertising system, we need to measure both the quantity and the quality of user contributions, and for each conversion event report the true "value" of the conversion. To achieve this aim, we set up Google Analytics to treat our site as an e-commerce website, and for each conversion we also report its value. Section 3 describes in detail our approach to quantifying the values of conversions.

When the advertising system receives fine-grained feedback about conversions and their value, it can improve the ad placement and display the ad to users who are more likely to participate and contribute high quality answers. (In our experiments, in Section 6, this optimization led to an increase in conversion rate from 20% to over 50%, within a period of one month, for a campaign that was already well-optimized.) For example, consider medical quizzes. We initially believed that identifying users with medical expertise who are willing to participate in our system would be an impossible task. However, thanks to tracking conversions and modeling the value of user contributions, AdWords started displaying our ad on websites such as Mayo Clicic and HealthLine. These websites are not frequented by medical professionals but by *prosumers*. These users are both competent and are much more likely than professionals to participate in a quiz that assesses their medical knowledge—often, this is exactly the type of users that a crowdsourcing application is looking for.

3. MEASURING USER CONTRIBUTIONS

In order to understand the contributions of a user for each quiz, we need first to define a measurement strategy. Measuring the user contribution using just the number of answers is problematic, as it does not consider the quality of the submissions. Similarly, if we just measure the quality of the submitted answers, we do not incentivize participation. Intuitively, we want users to contribute high quality answers, and also contribute many answers. Thus, we need a metric that increases as both quality and volume increase.

Information Gain: To combine both quality and quantity into a single, principled metric, we adopt an information-theoretic approach [36, 31]. We treat each user as a "noisy channel," and measure the total information "transmitted" by the user during her participation. The information is measured as the *information gain* contributed for each answer, multiplied by the total number of answers submitted by the user; this is the total information submitted by the user. More formally, assume that we know the probability q that the user answers correctly a randomly chosen question of the quiz. Then, the information gain $IG(q, n)$ is defined as:

$$IG(q, n) = H(1/n, n) - H(q, n) \qquad (1)$$

[2] https://adwords.google.com

[3] http://www.google.com/analytics

where n is the number of multiple choices in a quiz question. We use $H(q,n)$ to define the entropy[4] for an answer:

$$\begin{aligned} H(q,n) &= -\left(q\cdot\log(q)+\sum_{i=1}^{n-1}\left(\frac{1-q}{n-1}\right)\cdot\log\left(\frac{1-q}{n-1}\right)\right) \\ &= -q\cdot\log(q)-(1-q)\cdot\log\left(\frac{1-q}{n-1}\right) \qquad (2) \end{aligned}$$

When $q = 1$ (user always gives perfect answers), then $H(q,n) = 0$ (i.e., no uncertainty), and if $q = 1/n$ (user selects randomly from the n possible answers) then $H(q,n) = \log(n)$.

Information Gain under Uncertainty: In our environment, the quality q of a user is unknown. In fact, the goal of Quizz is to *estimate* q for each user, by asking the users to answer a set of quiz questions. We can try to approximate q with the ratio $q = \frac{a}{a+b}$, where a is the number of correct and b is the number of incorrect answers for the user, but we face the problem of sparse data, especially during the early stages of the quiz when $a + b$ is relatively small.

Due to the uncertainty in measuring the exact quality of each user, we introduce a Bayesian version of the information gain metric. Specifically, we explicitly acknowledge the uncertainty of our measurements, and we treat the estimate of q as a distribution, and not as a point estimate. The expected information gain when q is a random variable, we have:

$$E[IG(q,n)] = \int_{q=0}^{1} Pr(q)\cdot IG(q,n)\,dq \qquad (3)$$

In our system, we assume that q is constant across questions and latent.[5] However, we observe the number of correct answers a from the user; when q is constant, a follows a binomial distribution. We use the vanilla Bayesian estimation strategy [14] for estimating the probability of success q in a binomial distribution. We set $Beta(1,1)$ (i.e., the uniform distribution), as the conjugate prior and then $Pr(q)$ is a Beta distribution.[6] After the user submits a correct and b incorrect answers, $Pr(q)$ follows the $Beta(a+1,b+1)$ distribution:

$$Pr(q) = q^a\cdot(1-q)^b\frac{1}{B(a+1,b+1)} \qquad (4)$$

with $B(x,y)$ being the Beta function. After some algebraic manipulations, we have:

$$\begin{aligned} E[IG(a,b,n)] =& \log(n) - \frac{b}{a+b}\cdot\log(n-1) - \Psi(a+b+1) \\ &+ \frac{a\Psi(a+1)+b\Psi(b+1)}{a+b} \qquad (5) \end{aligned}$$

where $\Psi(x)$ is the digamma function. Figure 4 shows how $E[IG(a,b,n)]$ changes for different number of answers and for workers of varying competence. Following the same process, we also compute the variance of the information gain:

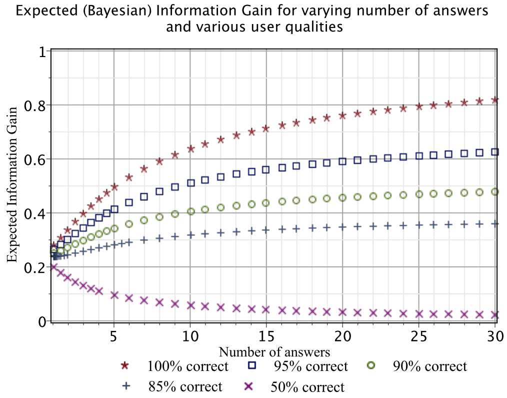


Figure 4: The expected (Bayesian) information gain values, for varying number answers, and various user qualities, when the number of available answers for each question $n = 2$.

$$Var[IG(q,n)] = \int_{q=0}^{1} (IG(q,n))^2\cdot Pr(q)\,dq - (E[IG(q,n)])^2$$

In the Appendix we list the detailed form of $Var[IG(q,n)]$. In the next section, we discuss how we use these measurements to optimally decide between assessing the user's knowledge and collecting new judgments.

4. EXPLORATION / EXPLOITATION

So far, we have described the setting where the user arrives and starts participating by answering quiz questions. Using the information gain metric, described in the previous section, we can estimate the amount of information that we can extract from a user if we ask a *collection* question, with an unknown (to us) answer. However, our goal is not just to estimate how much information we *could* get, but actually acquire new knowledge from the user. This creates a natural exploration-exploitation tradeoff. We can choose to "explore" how competent is the user, asking *calibration* questions, getting increasingly higher confidence about the user's competence on a topic; or we can try to "exploit," asking *collection* questions.

To formalize our decision making, we assume that the decision on whether to explore or exploit depends only on the current quiz that the user is solving and the current state of the user, which can be described by the number of correct answers a, the number of incorrect answers b, and the number of times c that we asked a collection question. Given the state vector $\langle a,b,c\rangle$ of the user, we use a Markov Decision Process (MDP) to select the next action to take, based on the following considerations.

- **User dropping out:** At any point, the user may opt to abandon the application. When the users drops out, we do not obtain any additional utility; therefore an optimal set of actions should try to steer the user towards states with high probability of "survival." (As we will see in Section 6, the probability of abandonment increases when the user gives incorrect answers to the quiz questions, and when the user does not receive feedback about the correctness of the submitted

[4] Note that the user can select among n possible answers, and we assume that the error probabilities are uniformly distributed among the $n-1$ incorrect answers, each being selected with probability $\frac{1-q}{n-1}$.

[5] In the future, we can use Item Response Theory [12] and allow each question to have its own q value.

[6] Alternatively, we can use a mixture of Beta priors to encode better our prior knowledge about the distribution of user qualities [27].

Algorithm 1: ComputeUtility(a, b, c, $U^{past}_{question}$, l)

```
Data: Correct answers $a$, Incorrect answers $b$, Unknown answers $c$, Question utility $U^{past}_{question}$, Horizon limit $l$
Result: Utility for all actions, Optimal next action
begin
    if $l < 0$ then
        return 0 // Reached the limit of computing horizon
    end

    $\gamma = Pr(survive|\langle a, b, c\rangle)$ // The (conditional) probability that the user will answer the served question.
    $eig = E[IG(a, b, n)]$ // Expected information gain
    $sig = \sqrt{Var[IG(a, b, n)]}$ // Standard deviation of information gain
    $U^{now}_{question} = eig - sig$ // The estimate of information gain for a question at the $\langle a, b, c\rangle$ state

    /* Utility estimation for a collection question                                   */
    $U^{now}_{coll} = U^{now}_{question}$ // If we ask a collection question, we get $eig - sig$ extra utility
    $U^{future}_{coll} = ComputeUtility(a, b, c + 1, U^{now}_{question}, l - 1)$ // Utility from future steps
    $U_{coll} = \gamma \cdot (U^{now}_{coll} + U^{future}_{coll})$ // Total utility of asking a collection question

    /* Utility estimation for a calibration question                                  */
    $q = (a + 1)/(a + b + n)$ // Probability of user answering correctly a calibration question
    /* Utility if the user answers correctly                                          */
    $U^{now/corr}_{question} = E[IG(a + 1, b, n)] - \sqrt{Var[IG(a + 1, b, n)]}$ // Information gain, after a correct answer
    $U^{now/corr}_{calib} = c \cdot (U^{now/corr}_{question} - U^{past}_{question})$ // Revise information gain for all $c$ previously-asked *collection* questions.
    $U^{fut/corr}_{calib} = ComputeUtility(a + 1, b, c, U^{now/corr}_{question}, l - 1)$ // Utility from future steps, after a correct answer
    /* Utility if the user answers incorrectly                                        */
    $U^{now/incorr}_{question} = E[IG(a, b + 1, n)] - \sqrt{Var[IG(a, b + 1, n)]}$ // Information gain, after an incorrect answer
    $U^{now/incorr}_{calib} = c \cdot (U^{now/incorr}_{question} - U^{past}_{question})$ // Revise information gain for all $c$ previously-asked *collection* questions.
    $U^{fut/incorr}_{calib} = ComputeUtility(a, b + 1, c, U^{now/incorr}_{question}, l - 1)$ // Utility from future steps, after an incorrect answer

    $U_{calib} = \gamma \cdot (q \cdot (U^{fut/corr}_{calib} + U^{now/corr}_{calib}) + (1 - q) \cdot (U^{fut/incorr}_{calib} + U^{now/incorr}_{calib}))$ // Total utility of calibration

    if $U_{calib} > U_{coll}$ then
        Action = Ask calibration question
    else
        Action = Ask collection question
    end
    return $\{U_{calib}, U_{coll}, U^{now}_{question}\}$, $Action$
end
```

answer.) In our application, we estimate the probability $Pr(survive|\langle a, b, c\rangle)$ based on the empirically-observed "lifetimes" of users, using a non-parametric kernel-density estimator with Gaussian smoothing.[7] (See Figure 6.)

- **Ask a collection question:** When we select to ask a collection question, there are two components for the utility that the Quizz system receives. Namely, there is immediate utility of getting information about the potential answer for the question, and there is utility that we will accumulate from the future actions of the user. The former utility is equal to the expected information gain for the user given his current state vector $\langle a, b, c\rangle$ (see Equation 3). However, we want to be more pessimistic about the utility estimates and assign more value to learning about the user competency. Following the "value of learning" approach [24], we therefore set the reward to $U^{now}_{question} = E[IG(q, n)] - \sqrt{Var[IG(q, n)]}$, in order to encourage our application to learn more about the user before asking her to contribute new knowledge. The utility for the future steps U^{future}_{coll} is the utility for the state vector $\langle a, b, c{+}1\rangle$, as we asked one more collection question.

- **Ask a calibration question:** When we select to ask a calibration question, we are trying to learn more about the competence of the user on the specific topic of the quiz. When presented with a question, the user may give either a correct or incorrect answer, which will lead to a revision of the $E[IG(q, n)]$ and $Var[IG(q, n)]$ metrics. Although we do not get directly a utility by asking a collection question, the revised estimate of $U^{now}_{question}$ applies to all c previously asked collection questions. Therefore, the $U^{now}_{calib} = c \cdot \left(U^{now}_{question} - U^{past}_{question}\right)$ where $U^{past}_{question}$ is the previous estimate of the utility of a collection question. Furthermore, the utility of future steps, is the stochastic sum of two possible forward paths: The utility $U^{correct}_{calib}$ when the user gives the correct answer (with probability $q = (a + 1)/(a + b + n)$), and the utility $U^{incorrect}_{calib}$ when the user gives an incorrect answer.

[7] We use the KernSmooth package in R.

Algorithm 1 describes the implementation of this MDP. One immediate concern with this formulation is that the recursive algorithm definition points to states in the future, and hence the total reward could potentially be infinite. This concern is alleviated if we assume that the information gain in each step in bounded, and that the probability of survival $\gamma = \max\{Pr(survive|\langle a, b, c\rangle)\} < 1$. We know that the maximum information gain derived from a single question is $\log(n)$ and there is always a non-zero probability that the user will abandon the application. Therefore, the total utility that can be extracted from a single user is bounded by $\sum_i \gamma^i \cdot \log(n) \leq \frac{\log(n)}{1-\gamma}$.

Another problem that arises with a recursive definition that points to future states is that the computational estimation becomes harder. Classic dynamic programming solutions assume a setting of backwards induction, where the recursion eventually leads to some initial state that has a known utility (e.g., the recursive computation of $U(\langle a, b, c\rangle)$ depends on $U(\langle a', b', c'\rangle)$ with $a' \leq a, b' \leq b, c' \leq c$). However, in our setting we have a forward induction, making the computation of recursion challenging. To allow the recursive computation to complete, we introduce a limited execution horizon for the recursion [29]: once the recursion has exceeded that level of depth, we stop the computation and return. To find out whether the algorithm converges, we run the algorithm iteratively with increasing horizon, until observing that the actions and utility calculations converge. Empirically, the algorithm converges faster when the survival probabilities $Pr(survive|\langle a, b, c\rangle)$ become smaller.

5. ENGAGEMENT INCENTIVES

In the previous section, we discussed how we can alternate between "exploration" and "exploitation" in order to assess the user's competence and collect new information, respectively. A key requirement for the algorithm to work effectively is to have a reasonable level of participation from the users: if a user submits just a couple of answers, we cannot effectively assess the user's competence or reliably collect new information.

Therefore, a key component of Quizz is the ability to continuously run experiments with various incentive mechanisms, that are trying to incentivize users to continue participating. Based on theories of intrinsic motivation [25], we implemented a variety of incentives, with the goal of prolonging the participation of competent users, while gently discouraging the non-knowledgeable users from submitting low-quality answers. Specifically, we tried the following options:

- **Feedback for submitted answer:** We experimented with various types of feedback that we give back to the users. We tried giving *no feedback*, *saying whether the answer was correct or not*, and *showing the correct answer*. The basic hypothesis is that immediate *performance feedback* [25] should motivate competent users to continue participating, and potentially incentivize low-performing users to try to improve their performance.

- **Displaying scores:** We experimented with displaying different types of scores to the user. We displayed the *percentage of correct answers*, the *total number of correct answers*, and a score based on the *information gain*, and combinations thereof.

- **Displaying crowd performance:** We experimented with showing the *crowd performance* on each question (i.e., how many users answered the question correctly). We hypothesized that knowing how other users perform is going to increase the user effort.

- **Leaderboards:** We experimented with showing the ranking of the user compared to other participants. Our hypothesis was that users will modify their behavior [2, 11] striving to reach one of the top leaderboard positions.

6. EVALUATION

6.1 Metrics

In order to evaluate our system, we used the following metrics to quantify the level of user engagement and the quality of their contributions.

- **Conversion rate:** We define conversion rate as the fraction of users who answered at least one quiz question, after clicking one of our ads (cf. Section 2). We use this metric to measure the effectiveness of our advertising.

- **User lifetime:** We examine the number of (correct and incorrect) answers submitted by the users. We use this metric mainly to understand the effect of the various engagement incentives.

- **Total information gain:** We measure the expected information gain of each user using Equation 3, and multiply this value by the total number of answers submitted by the user. The result is the total information that we received from the user.

- **Monetary cost per correct fact:** We measure the total cost required to verify a fact at the 90%, 95%, and 99% estimated accuracy. To compute the cost for various levels of accuracy, we use the fact that if we know the quality of a contributor, we can estimate the required redundancy to reach the desired level of confidence [35, 15]. For example, if we have two users that are 90% accurate and we pay a cost of \$0.10 per contributed answer, we need one such worker to verify a fact at 90% accuracy (i.e., cost \$0.10 at 90% accuracy), and approximately two such workers to verify a fact at the 99% (i.e., cost \$0.20 at 99% accuracy). To evaluate the correctness of the answers submitted by the users and the corresponding capacity of the system, we used questions with answers that have been pre-validated by multiple, trusted human judges. The costs are calculated based on the total advertising expenditure for attracting the users to the Quizz site, broken down by quiz.

6.2 Capacity and cost analysis

Based on our measurements for September 2013, the conversion rate for the Quizz application was an average of 34.60%, resulting in a total of 4,091 engaged users out of 11,825 users that visited the application (having clicked an ad). The conversion rate increased steadily over time, starting at around 20% in the beginning of the month, and reaching a high of 51.25% on September 30th. (As we will

Quiz	Users	Answers	Cost	Capacity/User			Cost/Answer		
				@99%	@95%	@90%	@99%	@95%	@90%
Disease Causes	414	7,644	$51.13	3.75	4.83	6.49	$0.07	$0.05	$0.04
Disease Symptoms	569	11,088	$12.51	3.30	4.25	5.71	$0.02	$0.01	$0.01
Treatment Side Effects	605	5,044	$46.38	1.22	1.57	2.12	$0.13	$0.10	$0.07
Artist and Albums	310	1,548	$21.56	0.88	1.13	1.52	$0.16	$0.13	$0.09
Latest Album	522	2,588	$20.70	0.95	1.23	1.65	$0.09	$0.07	$0.05
Artist and Song	925	5,285	$236.26	0.96	1.23	1.66	$0.54	$0.42	$0.31
Film Directors	412	2,250	$16.49	1.19	1.54	2.07	$0.07	$0.05	$0.04
Movie Actors	337	2,189	$36.14	0.96	1.24	1.66	$0.22	$0.18	$0.13
Average	*512*	*4,704*	*$55.15*	*1.65*	*2.13*	*2.86*	*$0.16*	*$0.12*	*$0.09*

Table 1: Answers that can be collected per user and corresponding cost at 90%, 95%, and 99% accuracy levels. The capacity per user metric is the number of collection questions that can be answered using the Quizz system, at different levels of accuracy. Cost is the amortized cost per question. All campaigns run with the conversion optimizer enabled, and a target cost-per-conversion of $0.10.

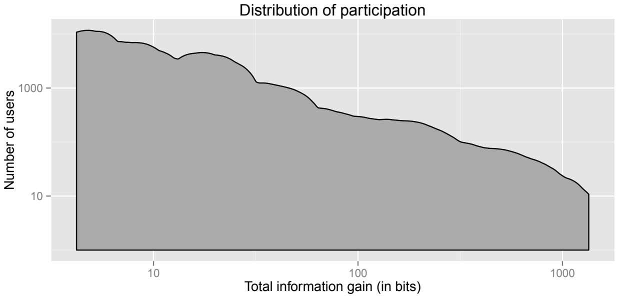


Figure 5: Distribution of participation. The vertical axis is the the number of users (in log scale), the horizontal axis is the of the total information gain.

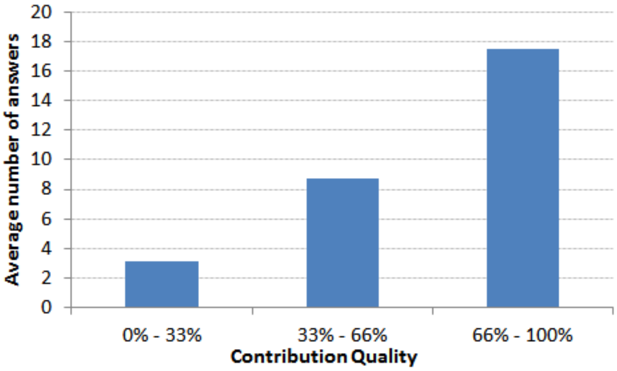


Figure 6: Lifetime of users according to their contribution quality.

discuss below, this is due to the continuous optimization from the conversion optimizer).

In our experiments we used eight different quizzes on various topics, and we report in Table 1 the detailed results in terms of user recruitment, cost, and the capacity of the system in providing answers to new questions at different accuracy levels. On average, we paid $0.16 to validate a fact at the 99% accuracy level, and $0.09 to validate a fact at the 95% accuracy level. An interesting outlier is the “Artist and Song” quiz, which ended up having significantly higher costs per collected answer compared to the other quizzes. In this case, effectively we observed a failure of the advertising system to identify a group of users that would be knowledgeable of *all* the artists and songs that appeared in the quiz: given the diversity of music genres in that quiz, it was difficult to find music fans that have *detailed* knowledge of all the song titles of various artists across multiple genres. (However, it *was* possible to find music fans that have knowledge of at least the album names of different artists, an admittedly easier task.)

6.3 User contributions and self-selection

In terms of a per-user contribution, Figure 5 shows the distribution of total information gain across the participating users, and Figure 6 shows the lifetime of users as a function of their quality. As expected, many users come, submit a few answers, and then leave. These are the “head” users; although they do contribute some useful signal, they do not generate a great “return on investment.” Figure 7 further illustrates that the users that submit large number of answers also tend to submit more correct answers than incorrect. This means that the users who are competent about the topic submit more and more answers, while the ones who cannot answer the quiz questions correctly, drop out. This is an illustration of the benefit of unpaid users: there is little incentive for unpaid users to continue participating when there is no monetary reward and they are not good at the task. (We present a more detailed comparison with paid crowdsourcing in Section 6.7.)

6.4 The effect of targeting in advertising

A major hypothesis of our work is that the targeting system of existing advertising networks can be leveraged in order to identify competent users, who are willing to contribute new knowledge by answering quiz questions. We observe that users recruited through advertising are knowledgable and willing to contribute. However, it is not immediate obvious whether the positive result is due to targeting, or is simply the effect of bringing more users to the application.

In order to disentangle the effects of advertising and targeting, we ran two different advertising campaigns that both directed users to the same quiz. Both campaigns had the same budget, same ad creatives, same bidding settings, and their only differences were (1) the keywords used for the bidding and (2) the use of feedback to the advertising system.

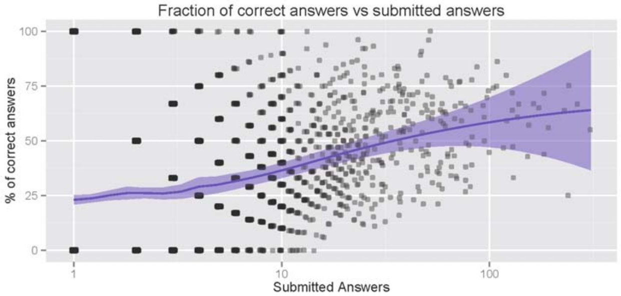
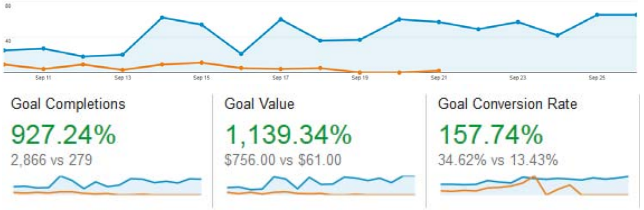


Figure 7: Quality of submissions as a function of user participation. (Intensity of dot color corresponds to the number of users represented by the dot; fitted line computed using LOESS.) Knowledgeable users self-select and to continue submitting answers, leading to significant increase of answer quality for heavy participants. Low-performers drop out voluntarily without submitting many answers.



Goal Completions
927.24%
2,866 vs 279

Goal Value
1,139.34%
$756.00 vs $61.00

Goal Conversion Rate
157.74%
34.62% vs 13.43%

Figure 8: Comparison of conversions for targeted vs. untargeted ad campaign (screenshot from Google Analytics; the upper plot shows the information gain for the two campaigns as a function of time; blue: targeted, yellow: untargeted). The targeted campaign generates 9.2x more conversions, as well as higher-quality answers.

In the targeted campaign, we used keywords related to the topic of the quiz; for the untargeted campaign we used the keywords from all the quizzes available in the Quizz system. Also, in the untargeted campaign, we did not send feedback about conversions to avoid providing targeting information.

Figure 8 shows the results. While the number of visitors was roughly the same for the two campaigns, the targeted campaign had 3x higher conversion rate (34.62% vs. 13.43%). Furthermore, among the participating ("converted") users, the number of questions answered per user was 3x higher for the users who arrived from the targeted campaign. Thus, the cumulative difference was over 9.2x more answers obtained through the targeted campaign compared to the untargeted one (2866 answers vs. 279). Finally, the answers contributed by the users from the targeted campaign were of higher quality than the answers from the untargeted campaign: the total information gain for the targeted campaign was 11.4x higher than the total information gain for the untargeted one (7560 bits vs. 610 bits), indicating a higher user competence, even on a normalized, per-question basis.

6.5 The effect of using conversion optimizer

After verifying that targeting and feedback indeed improve the results, we wanted to examine the effect of using the *conversion optimizer*. While traditional ad campaigns usually optimize for clicks, the conversion optimizer of Google AdWords offers the option to optimize for the total "value" of the conversions (in our case, for the total information gain). To examine the usefulness of the conversion optimizer in our setting, we again run two otherwise-identical ad campaigns: one being optimized for clicks, and the other being optimized for conversions.

The conversion rate increased by 30% when using the conversion optimizer (from 29% to 39%). In addition to that, the number of submitted answers went up by 42% (1683 vs. 1183), and the total information gain went up by 63% (4690 bits vs. 2870 bits). Furthermore, as Section 6.2 discusses, the optimization is ongoing and the conversion rate continues to go up even at the time of this writing. This automatic and continuous optimization of the process illustrates the benefits of leveraging existing, publicly available advertising platforms to improve the efficiency of crowdsourcing applications.

Variable: Total	Coefficients	Significance
showCorrect	0.142475	***
showMessage	-0.008423	
showPercentageCorrect	-0.003646	
showTotalCorrect	0.085231	***
showScore	0.093463	***
showCrowdAnswers	0.025502	*
showPercentageRank	-0.074008	***
showTotalCorrectRank	-0.006259	
Variable: Correct		
showCorrect	0.13936	***
showMessage	0.002874	
showPercentageCorrect	0.010314	
showTotalCorrect	0.085838	***
showScore	0.062361	***
showCrowdAnswers	0.041233	*
showPercentageRank	-0.101775	***
showTotalCorrectRank	-0.007062	
Variable: Score		
showCorrect	0.204104	***
showMessage	0.024464	**
showPercentageCorrect	0.023831	***
showTotalCorrect	-0.022065	***
showScore	0.040387	***
showCrowdAnswers	0.098557	***
showPercentageRank	-0.048658	***
showTotalCorrectRank	-0.016987	***

Table 2: The effect of various mechanisms in incentivizing users to submit more answers *Total*, more correct answers *Correct*, and to improve their total information gain *Score*. The coefficients were computed by a Poisson regression model (*: 0.1% significance, **: 1% significance, *: 5% significance). Given that $e^x \approx 1+x$ for small values of x, a coefficient of 0.1 means that we observe a 10% improvement, while a coefficient of -0.1 means that we observe a 10% decrease in performance.**

6.6 The effect of engagement incentives

To analyze the effect of the various incentive mechanisms, we examined how the different experimental conditions assigned to the users affected their participation and their overall contributions. To this end, we examined the effect of the various incentives on three variables of interest: the *total* number of submitted answers, the number of *correct* answers, and the (total information gain) *score* of the user. Since the dependent variables are always positive and behave like "count data" we ran a Poisson regression, with eight binary variables as dependent variables, where each of these variables corresponded to the presence (or absence) of an experimental condition. Specifically, we present results for the following incentive mechanisms:

- **showCorrect**: Show the correct answer.
- **showCrowdAnswers**: Show the percentage of other users who answered the question correctly.
- **showMessage**: Show whether the given answer was correct.
- **showPercentageCorrect**: Show the percentage of submitted answers (for that user) that were correct.
- **showTotalCorrect**: Show the total number of correct answers submitted by that user.
- **showScore**: Show the total information gain for the user (shown as a score).
- **showPercentageRank**: Show the position of the user in the leaderboard, ranked by percentage of correct answers.
- **showTotalCorrectRank**: Show the position of the user in the leaderboard, ranked by the total number of correct answers submitted.

Table 2 summarizes the results, and shows the coefficients computed for each mechanism by the regression model. Showing the correct answer (showCorrect) has the strongest impact in increasing participation, as it has strong positive effect across all three dependent variables, indicating that users want to know what the correct answer is. Interestingly, knowing whether they were correct or not (showMessage) does not have a similarly strong effect. These results indicate that users may be more interested in *learning* about the topic rather than just knowing whether they answered correctly.

Experimenting with the performance-related incentives (i.e., showPercentageCorrect, showTotalCorrect, showScore) generated some interesting observations. Showing the percentage of correct answers did not have a statistically significant effect in terms of answer counts, but had a slightly positive effect in the total information gain. Showing the total number of correct answers generated an interesting effect: while both total and correct answers went up, the total information gain was affected negatively. It appears that non-competent users were also positively influenced to participate more, leading to a decrease of the overall answer quality. Not surprisingly, when we show the total information gain as a score to the user, this effect disappears, and we observe positive outcomes across all variables.

Finally, the competitive incentives (i.e., showCrowdAnswers, showPercentageRank, showTotalCorrectRank) demonstrated an interesting behavior of the users: knowing the performance of other users has a positive effect in the participation, which indicates that users are interested in how they fare against other users. However, displaying leaderboards had a generally negative effect across all variables.

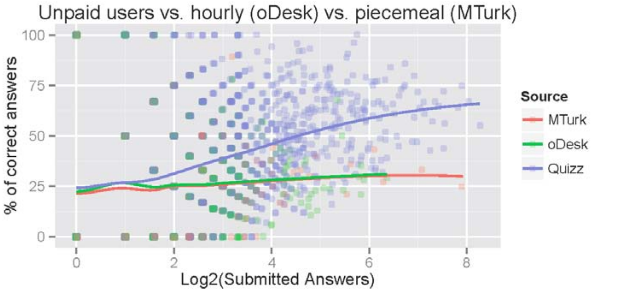


Figure 9: Comparison of unpaid users vs. common payment schemes (hourly and piecemeal payments). Paid workers are not experts in the presented topic, and their quality is barely above random, despite the bonus incentives for good performance; furthermore, the monetary reward incentivizes low-perfoming paid workers to continue participating.

Interestingly enough, if we examine the effect of leaderboards in the early stages of the application, we see a very strong *positive* effect in terms of participation. Our hypothesis to explain these contradicting observations is the following. Early on, the leaderboards are relatively sparsely populated and it is relatively easy for users to go up and reach some of the top positions. However, as more and more users participate, the achievements of the top users are difficult to match, effectively discouraging users from trying harder. To test our hypothesis, we ran a small experiment, where the leaderboard was computed based on the participation from last week, as opposed to showing an all-time leaderboard. The results indicated that the "last-week" leaderboard with fewer and less impressive achievements has indeed had a *positive* effect on participation, compared to the "all-time" leaderboard. This indicates that users are motivated by the potential for achievement, and by showing the users that they can reach an achievement in a relatively easy manner can help with participation.

6.7 Comparison with paid crowdsourcing

Finally, we wanted to compare the performance of our approach against a pure paid-crowdsourcing setting. To this end, we hired workers through Mechanical Turk, and paid them 5 cents per question (i.e., piecemeal payment), with an extra bonus that depended on their total score (information gain) at the end. Similarly, we hired workers via oDesk, paid them on an hourly basis (ranging from \$5/hr to \$15/hr, depending on their asking price), and we also indicated that they will receive an additional payment based on their overall score. Figure 9 summarizes the results. Our key observation is that the workers hired through paid crowdsourcing platforms are usually not experts in the topic of the quiz, and are therefore not sufficiently knowledgeable to provide high quality answers. However, unlike the unpaid workers, the paid workers have an obvious monetary incentive to continue working, and so we did not observe the self-selection dropout effect for the paid workers. The paid workers continue submitting low-quality answers, and this finding is similar with both piecemeal and hourly payments.

While there are some compeqtent workers among the paid participants, the total information gain from the competent paid workers is still significantly lower than the information gain from unpaid users, resulting in significantly lower capacities. The *best* paid worker had a 68% quality for the quiz, and submitted 40 answers, resulting in an equivalent capacity of 13 answers at 99% accuracy, or 23 answers at 90% accuracy. To match the performance of the unpaid users, the worker should be paid 5 cents per question, or $3/hr, taking into account that the average time per question *for the paid users* is one minute. (For comparison, unpaid users are much faster and typically give an answer within 10 seconds, signaling that they are already knowledgeable about the topic of hte quiz and they do not perform research to answer the questions.) Given that all other workers demonstrated worse metrics, it is clear that unpaid, volunteer users dominate. A potential solution is to experiment with negative incentives (e.g., "you will not get paid unless you achieve this quality score"), keeping away the low perfomers, and keeping just the top workers. However, it is not clear how we can reach these high-quality workers in a labor market, other than by posting the task and then hoping that the competent workers will participate. Potentially, labor marketplaces can employ targeting schemes, similar to the ones we implemented using online advertising, but today we are not aware of any marketplace offering such functionality.

7. RELATED WORK

Quizz crowdsources the acquisition of knowledge by asking users to participate in thematically-focuses quizzes, which contain also "collection" questions with no known answer. ReCAPTCHA [34] is close conceptually, as it asks users to type two digitized words, out of which one is known and the other is unknown, which is similar to our calibration and collection questions, respectively. In terms of use of advertising for recruiting users, Hoffman et al. [17] use advertising to attract participants for a Wikipedia-editing experiment; however there was no discussion or experiments with targeting, or with optimizing the ad campaigns for maximizing the user contributions.

Recent work [2, 11] built models on how badges and leaderboards should be designed to engage users and steer their behavior towards actions that are beneficial for the system. Our work empirically tests some of these models, and our experimental results dovetail the suggestions of these models. Other models of user engagement have examined what metrics and measurments capture the user level of engagement [23, 5, 6, 10, 26]. Our analysis of engagement focuses mainly on web analytics measurements, without trying to interact further with the participating users, although this is a promising direction for future work.

In our work, we explicitly assess the competence of users with calibration questions. Alternatively, we can use unsupervised techniques for estimating the competence of users, through redundancy. Dawid and Skene [8] presented an EM algorithm to estimate the quality of the participants in the absence of known ground truth, and a large number of recent papers examined the same topic [30, 39, 37] improving significantly the state of the art. Being closer to our work, Kamar et al. [21] also use a Markov Decision Process, in order to decide whether the answers provided by a user are promising enough to warrant a hiring decision. In the future, we plan to use these algorithms for quality inference together with our exploration/exploitation approach, to decide optimally how to combine assessment with knowledge acquisition. A key challenge is being able to provide immediate feedback to the users, when the questions have no certain answer.

Optimal acceptance sampling plans in quality control [9, 38, 7, 32] is another related line of work. The purpose of acceptance sampling is to determine how much to sample a production line, in order to decide whether to accept or reject a production lot. The key difference with our setting is the limited lifetime of the users (as opposed to the significantly higher production capacity in industrial production), and our planning needs to be much more dynamic than in the most use cases of acceptance sampling.

8. CONCLUSIONS

We presented a model for targeting and engaging *unpaid* users in a crowdsourcing application. We demonstrated how to use existing Internet advertising platforms to identify niche audiences of competent users for the task at hand, and we showed that using publicly available ad-optimization tools can result in significant improvements in the effectiveness of the process. Currently, our application has a 50% conversion rate for every ad click, and the cost per answer drops systematically over time, as the advertising system learns to identify competent users that are likely to be high contributors. The engagement of unpaid users alleviates concerns about the incentives of paid users, who are not always well-aligned with the goals of the crowdsourcing application. Furthermore, our algorithms and controlled real-life experiments with over ten thousand users illustrate how to setup incentive mechanisms in practice to engage users and extend their "lifetime" in the system. Finally, our experiments indicate that even though there are costs associated with advertising, the quality-adjusted costs are on par with those of paid crowdsourcing. (Moreover, for non-profits, engaging for example in *citizen science* efforts, there are ways to get a substantial advertising budget using programs such as "*Google Ad Grants for nonprofits*,"[8] which offers $10,000 per month in in-kind advertising budget.) We believe that our ad-based approach can form the foundation towards more predictable deployment and engagement of unpaid users in crowdsourcing efforts, combining the advantage of engaging unpaid users with the predictability of paid crowdsourcing.

Acknowledgments

We would like to thank Kevin Murphy and Chun How Tan for helpful discussions and suggestions.

APPENDIX

A. VARIANCE OF INFORMATION GAIN

When the quality q of a user is uncertain, then the information gain for each question is also uncertain. Under the assumption the probability q, that a user answers a question correctly, is the same across all questions, and that the prior is a uniform distribution, then the variance of the information gain distribution is given by:

[8] http://www.google.com/grants/

$$
\begin{aligned}
Var[IG(q,n)] = \log(n)^2 + \frac{2ab \cdot I_{ab}}{(a+b)(a+b+1)} \\
+\frac{a(a+1) \cdot J_a}{(a+b+1)(a+b)} + \frac{b(b+1) \cdot J_b}{(a+b+1)(a+b)} \\
+\frac{b(b+1)\log(n-1)^2}{(a+b)(a+b+1)} + \frac{2b \cdot \log(n-1) \cdot K_{ab}}{(a+b) \cdot (a+b+1)} \\
-\frac{2\log(n) \cdot \log(n-1) \cdot b}{a+b} - 2\log(n) \cdot E[IG(a,b,n)]
\end{aligned}
$$

$$
\begin{aligned}
J_a &= (\Psi(a+2) - \Psi(a+b+2))^2 \\
&\quad +\Psi_1(a+2) - \Psi_1(a+b+2) \\
J_b &= (\Psi(b+2) - \Psi(a+b+2))^2 \\
&\quad +\Psi_1(b+2) - \Psi_1(a+b+2) \\
I_{ab} &= (\Psi(a+1) - \Psi(a+b+2)) \cdot (\Psi(b+1) \\
&\quad -\Psi(a+b+2)) - \Psi_1(a+b+2) \\
K_{ab} &= (a+b+1)\Psi(a+b+1) \\
&\quad -(b+1)\Psi(b+1) - a\Psi(a+1)
\end{aligned}
$$

where $\Psi(x)$ is the digamma function, $\Psi_1(x)$ is the trigamma function, n is the number of options presented to the user, $a-1$ is the number of correct, and $b-1$ is the number of incorrect answers submitted by the user [3].

B. REFERENCES

[1] Abhishek, V., and Hosanagar, K. Keyword generation for search engine advertising using semantic similarity between terms. In *Proceedings of the ninth international conference on Electronic commerce* (2007), ACM, pp. 89–94.

[2] Anderson, A., Huttenlocher, D., Kleinberg, J., and Leskovec, J. Steering user behavior with badges. In *Proceedings of the 22nd international conference on World Wide Web* (2013), International World Wide Web Conferences Steering Committee, pp. 95–106.

[3] Archer, E., Park, I. M., and Pillow, J. W. Bayesian entropy estimation for countable discrete distributions. *CoRR abs/1302.0328* (2013).

[4] Ariely, D. *Predictably irrational: The hidden forces that shape our decisions*. HarperCollins, 2009.

[5] Attfield, S., Kazai, G., Lalmas, M., and Piwowarski, B. Towards a science of user engagement (position paper). In *WSDM Workshop on User Modelling for Web Applications* (2011).

[6] Baeza-Yates, R., and Lalmas, M. User engagement: the network effect matters! In *Proceedings of the 21st ACM international conference on Information and knowledge management* (2012), ACM, pp. 1–2.

[7] Berger, R. L. Multiparameter hypothesis testing and acceptance sampling. *Technometrics 24*, 4 (1982), 295–300.

[8] Dawid, A. P., and Skene, A. M. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Applied Statistics 28*, 1 (Sept. 1979), 20–28.

[9] Dodge, H. F. *Notes on the Evolution of Acceptance Sampling*. American Society for Quality Control, 1973.

[10] Dupret, G., and Lalmas, M. Absence time and user engagement: evaluating ranking functions. In *Proceedings of the sixth ACM international conference on Web search and data mining* (2013), ACM, pp. 173–182.

[11] Easley, D., and Ghosh, A. Incentives, gamification, and game theory: an economic approach to badge design. In *Proceedings of the fourteenth ACM conference on Electronic commerce* (2013), EC '13, ACM, pp. 359–376.

[12] Embretson, S. E., and Reise, S. P. *Item response theory*. Psychology Press, 2000.

[13] Fuxman, A., Tsaparas, P., Achan, K., and Agrawal, R. Using the wisdom of the crowds for keyword generation. In *Proceedings of the 17th international conference on World Wide Web* (2008), ACM, pp. 61–70.

[14] Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. *Bayesian data analysis*. CRC press, 2003.

[15] Genest, C., and Zidek, J. V. Combining probability distributions: A critique and an annotated bibliography. *Statistical Science 1*, 1 (1986), 114–135.

[16] Gneezy, U., and Rustichini, A. Pay enough or don't pay at all. *The Quarterly Journal of Economics 115*, 3 (2000), 791–810.

[17] Hoffmann, R., Amershi, S., Patel, K., Wu, F., Fogarty, J., and Weld, D. S. Amplifying community content creation with mixed initiative information extraction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2009), ACM, pp. 1849–1858.

[18] Horton, J. Online labor markets. *Internet and Network Economics* (2010), 515–522.

[19] Ipeirotis, P. Demographics of mechanical turk. Tech. rep., New York University, 2010. Available at http://ssrn.com/abstract=1585030.

[20] Joshi, A., and Motwani, R. Keyword generation for search engine advertising. In *Data Mining Workshops, 2006. ICDM Workshops 2006. Sixth IEEE International Conference on* (2006), IEEE, pp. 490–496.

[21] Kamar, E., Hacker, S., and Horvitz, E. Combining human and machine intelligence in large-scale crowdsourcing. In *AAMAS* (2012), International Foundation for Autonomous Agents and Multiagent Systems, pp. 467–474.

[22] Kuznetsov, S. Motivations of contributors to wikipedia. *ACM SIGCAS computers and society 36*, 2 (2006), 1.

[23] Lehmann, J., Lalmas, M., Yom-Tov, E., and Dupret, G. Models of user engagement. In *User Modeling, Adaptation, and Personalization*. Springer, 2012, pp. 164–175.

[24] Li, S.-M., Mahdian, M., and McAfee, R. P. Value of learning in sponsored search auctions. In *Internet and Network Economics*. Springer, 2010, pp. 294–305.

[25] Malone, T. W. Toward a theory of intrinsically motivating instruction. *Cognitive science 5*, 4 (1981), 333–369.

[26] McCay-Peet, L., Lalmas, M., and Navalpakkam, V. On saliency, affect and focused attention. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2012), ACM, pp. 541–550.

[27] Murphy, K. P. *Machine learning: a probabilistic perspective*. The MIT Press, 2012.

[28] Nov, O. What motivates wikipedians? *Communications of the ACM 50*, 11 (2007), 60–64.

[29] Puterman, M. L. *Markov decision processes: Discrete stochastic dynamic programming*, vol. 414. Wiley. com, 2009.

[30] Raykar, V. C., Yu, S., Zhao, L. H., Valadez, G. H., Florin, C., Bogoni, L., and Moy, L. Learning from crowds. *The Journal of Machine Learning Research 99* (2010), 1297–1322.

[31] Robertson, S., Vojnovic, M., and Weber, I. Rethinking the esp game. In *CHI'09 Extended Abstracts on Human Factors in Computing Systems* (2009), ACM, pp. 3937–3942.

[32] Schilling, E. G. *Acceptance Sampling in Quality Control*, vol. 42. CRC PressI Llc, 1982.

[33] Von Ahn, L., and Dabbish, L. Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (2004), ACM, pp. 319–326.

[34] Von Ahn, L., Maurer, B., McMillen, C., Abraham, D., and Blum, M. Recaptcha: Human-based character recognition via web security measures. *Science 321*, 5895 (2008), 1465–1468.

[35] Wang, J., Ipeirotis, P., and Provost, F. Quality-based pricing for crowdsourced workers. Tech. rep., New York University, 2013. Available at papers.ssrn.com/abstract=2283000.

[36] Waterhouse, T. P. Pay by the bit: an information-theoretic metric for collective human judgment. In *Proceedings of the 2013 conference on Computer supported cooperative work* (2013), ACM, pp. 623–638.

[37] Welinder, P., Branson, S., Belongie, S., and Perona, P. The multidimensional wisdom of crowds. *Advances in Neural Information Processing Systems 23* (2010), 2424–2432.

[38] Wetherill, G., and Chiu, W. A review of acceptance sampling schemes with emphasis on the economic aspect. *International Statistical Review/Revue Internationale de Statistique* (1975), 191–210.

[39] Whitehill, J., Wu, T.-f., Bergsma, J., Movellan, J. R., and Ruvolo, P. L. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in neural information processing systems* (2009), pp. 2035–2043.

[40] Yang, H.-L., and Lai, C.-Y. Motivations of wikipedia content contributors. *Computers in Human Behavior 26*, 6 (2010), 1377–1383.