Proceedings of the Fifth Conference on
Human Computation and Crowdsourcing
(HCOMP 2017)

Crowdsourcing Paper Screening in Systematic Literature Reviews

Evgeny Krivosheev,[1] **Fabio Casati,**[1,2] **Valentina Caforio,**[1] **Boualem Benatallah**[3]
[1] University of Trento, Italy
[2] Tomsk Polytechnic University, Russia
[3] University of New South Wales, Australia
{firstname}.{lastname}@unitn.it, boualem@cse.unsw.edu.au

Abstract

Literature reviews allow scientists to stand on the shoulders of giants, showing promising directions, summarizing progress, and pointing out existing challenges in research. At the same time conducting a systematic literature review is a laborious and consequently expensive process. In the last decade, there have been several studies on crowdsourcing in literature reviews. This paper explores the feasibility of crowdsourcing for facilitating the literature review process in terms of results, time and effort, and identifies which crowdsourcing strategies provide the best results based on the budget available. In particular we focus on the screening phase of the literature review process and we contribute and assess strategies for running crowdsourcing tasks that are efficient in terms of budget and classification error. Finally, we present our findings based on experiments run on Crowdflower.

Introduction

A literature review is a form of scientific research (and of publication) that has a high impact on science and society (Sun et al. 2016). Reviews can take different forms and have different objectives (Grant and Booth 2009). The main distinction is between *systematic* approaches, where a specific process is defined before the review starts and is followed throughout the identification and analysis of relevant literature, and *non-systematic* ones, where authors do not follow a predefined method for selecting and analyzing literature.

Literature reviews, especially when systematic, provide scientific results and are at the heart of evidence-based approaches, with a potentially profound impact on society (Haidich 2010). Reviews are also very helpful in introducing newcomers to challenges and opportunities for research in a given area. Not surprisingly, they are among the most highly cited papers (a search we conducted over a few thousands papers on Scopus shows that the median number of citations for reviews vastly exceeds the median for papers in all areas of science).

Because of their importance and impact, the number of published reviews is rapidly growing (Wallace et al. 2013). This is particularly true for systematic reviews and meta-analyses, in the past popular mostly in the medical field but now widely adopted in all areas of science.

However, reviews are very time-consuming and effort-intensive. While there are no published statistics on the entire review process (from idea to publication) we are aware of, a study we are conducting with researchers from different fields points to durations of 6 months to 3 years from initial search to submission[1]. Review results should also be updated periodically, but again the effort for doing so often represents a barrier (Takwoingi et al. 2013).

In this paper we investigate the possibility of crowdsourcing specific aspects of systematic literature reviews. We focus specifically on identifying the in-scope papers after initial literature search, and we investigate if and how this phase can be sourced from the citizens, what are the best strategies for doing so, and what is the resulting quality and cost, both in general and compared with the case where the same phase is done by the research team (typically, the co-authors). This is a critical phase of a systematic review: not only is it time-consuming (several people work on it, and the combined person-month effort is of over two months), but it is also where risk of bias lies.

More specifically, we contribute i) a probabilistic model for reasoning over the problem, for tuning the parameters of crowdsourcing tasks to minimize errors, and for providing review authors with information of budget vs error trade-offs, and ii) a set of crowdsourcing strategies and algorithms that minimize the classification error as we vary the assumptions on the model and the model parameters. Both the model and the strategies descend from experiments run on *Crowdflower*[2] and are mindful of what we can actually achieve with some of the practical constraints of typical crowdsourcing platforms. Experiments on Crowdflower are also used, in addition to theory and simulation, to validate the results as well as to derive parameters for the typical population of workers for this kind of tasks.

Last but not least, experiments provided many insights on task design, such as how the problem should be framed to increase participation and reduce errors, as well as actual

[1] Published data in the healthcare domain indicate that the median time from the *final* literature search to publication in a systematic review is 61 weeks - with the additional problem that over time the list of candidate papers for inclusion becomes out of date and needs to be refreshed (Sampson et al. 2008). However the paper does not report on lag from initial search.

[2] www.crowdflower.com

pay scales considered acceptable by the community.

Background and Related Work

PRISMA and Systematic Reviews

Before presenting our approach we summarize methods and practices for systematic reviews. A systematic review follows a defined *process* and has transparency and clarity as its focal points throughout the whole procedure (Khan et al. 2003). This process usually includes (i) the definition of a research question in a clear, structured and unambiguous way; (ii) the identification of all relevant papers through a search strategy that stems from the research question and specifies inclusion and exclusion criteria; (iii) the critical assessment of the included studies; (iv) the data extraction and synthesis in a standardized form, possibly with statistical analysis (meta-analysis); (v) the interpretation of the findings and exploration of any risk for bias (Khan et al. 2003; Wright et al. 2007; Harris et al. 2014; Henderson et al. 2010).

With the objective of increasing the quality of systematic reviews and meta-analyses, the PRISMA statement (*Preferred Reporting Items for Systematic Reviews and Meta-Analyses*) was devised as a guideline to help authors report their reviews in a clear and consistent way (Moher et al. 2009). As an evolution from the QUOROM statement (Moher et al. 1999), PRISMA consists of a 27-items checklist enumerating the details to report and a flow diagram showing the phases of the selection process. Such statements are often required in any systematic review today and are essential in the medical field, where poorly reported reviews can potentially have an effect on people's health. Indeed, Clinical Practice Guidelines, i.e., "statements that include recommendations intended to optimize patient care", are "based on systematic reviews of evidence" and should "be based on an explicit and transparent process that minimizes distortions, biases, and conflicts of interest" (Steinberg et al. 2011). Therefore, omitted details and lack of transparency can make this process difficult and contribute to low-quality, misleading guidelines.

Crowdsourcing and Science

Crowdsourcing is being increasingly adopted as a tool for supporting research (Law et al. 2017). There are literally hundreds of *citizen science* projects that leverage crowdsourcing at one phase or another of the research, in all fields of science, from biology to astronomy to human sciences (Garneau et al. 2014; Swanson et al. 2015; Hennon et al. 2015; Lintott et al. 2008). The interest in citizen science has generated a growing body of research on various aspects of the process, from understanding how researchers perceive it (H. Riesch H 2014; Law et al. 2017), to the motivations behind citizens' participation (Eveleigh et al. 2014; Frey and Jegen 2001), as well as process and system design (Tinati et al. 2015).

While all aspects of citizen science research are somewhat interesting and related to this paper, one item of particular importance is the understanding of the conditions under which researchers are motivated to (or deterred from) adopting a crowdsourcing approach. A beautiful analysis of these aspects is provided in (Law et al. 2017) who underscore that one of the concerns is related to how *reviewers* perceive crowdsourcing in research. In other words, crowdsourcing is viable if i) the authors feel that is feasible and valuable for their specific research problem and ii) the authors perceive that reviewers will find it acceptable. This is relevant because literature reviews go through peer reviews and as such the process needs to be accepted by reviewers - and by the community.

The prior art also includes several "spot" attempts at adopting some form of crowdsourcing in literature reviews. These papers provided inspiration for us although in many cases they are initial, one-off, and relatively small experiments that do not study the variations of the results of crowdsourcing tasks in terms of its content and parameters, and that in general do not have sufficient statistical power to derive conclusions and guidelines.

Sun and colleagues (Sun et al. 2016) study the feasibility of leveraging crowd workers for extracting information related to interventions from papers abstracts in biomedical domains. The authors observe that giving more concrete examples in the instruction part can help workers be more aware of the purpose of a task. A platform for crowdsourcing narrative literature reviews is proposed by Weiss (Weiss 2016), along with insight about challenges appearing in systematic literature reviews in new domains. Nguyen et al. (Nguyen, Wallace, and Lease 2015) propose an active learning approach to solving the problem of deciding whether or not a paper is relevant to a review, with a mixed human+AI approach. The authors tried to achieve maximum performance at minimal cost by intelligently choosing between crowd users and domain experts to minimize the expected loss. They performed experiments on Amazon's Mechanical Turk to classify papers from four large datasets. For crowd evaluation, classification is performed through majority voting.

Ng and colleagues (Ng et al. 2014) ran a randomized pilot study aimed at exploring the accuracy of medical students in performing citation screening via four different modalities, namely a mobile screening application, paper printed with titles and abstracts, a reference management software and a web-based systematic review platform. Students were asked to say whether a list of papers were included or excluded from the scope of a review, based on a list of inclusion criteria. In case of insufficient information, participants could set papers as "unsure". Participants had never conducted a review, however they had some level of expertise in the field and had received some training in the development of critical appraisal skills, which differs widely from asking to a non-expert crowd to perform such a task.

Modeling and Classification

Extensive research, dating back to the 1700s, has addressed the problem of eliciting reliable labels from a crowd, coping with cheating behavior while keeping costs low (Karger, Oh, and Shah 2011b; Whitehill et al. 2009; Smyth et al. 1995; Karger, Oh, and Shah 2011a; Hirth, Hoßfeld, and Tran-Gia 2013; Liu, Ihler, and Steyvers 2013; Eickhoff and de Vries

Sign	Specification
$CP = \{p_1, ..., p_n\}$	set of candidate papers
l_{pj}	vote by worker j on paper p
cr	cost ratio
N_t	#test questions
N_l	#votes per worker (after passing test)
J	#votes per paper
S	scope of a review
UC	cost per vote
PPP	price per paper
z	proportion of cheaters
$\overline{a}_s$	expected accuracy of workers
θ_i	real proportion of papers *in* scope

Table 1: Notations used in the paper.

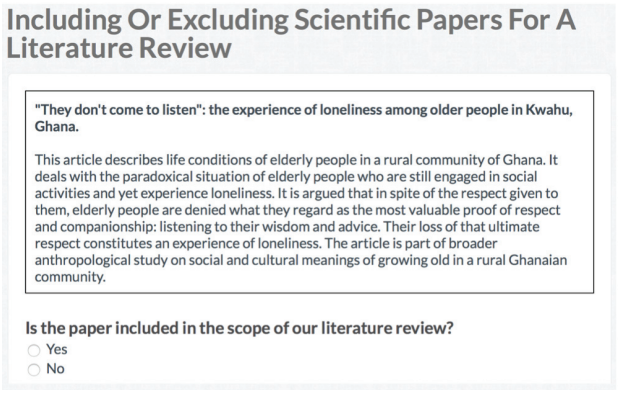

Figure 1: Example of scope-based screening task.

2013; Hirth, Hoßfeld, and Tran-Gia 2011).

One of the first scientists to study this was the Marquis of Condorcet. Condorcet, in his famous *Jury Theorem*[3] of 1785, discusses the probability of a group of persons taking, collectively, the correct classification decision. He shows that if the probability of a person taking the correct decision is greater than 0.5 and votes are independent of each other, then the probability of taking a correct majority decision grows with the number of participants and approaches 1 at the limit (this is, in fact, a direct consequence of the law of large numbers).

From there, a large body of work starting with (Dawid and Skene 1979) and then on to (Whitehill et al. 2009), estimating labeling in the presence of items of different difficulties, and (Liu and Wang 2012) who apply EM to labeling in the presence of confusion matrix inspire our approach. We also build on insights from Hirth and colleagues (Hirth, Hoßfeld, and Tran-Gia 2013), who discuss the problem of cost optimization providing information on which cheating detection and task validation algorithms to choose based on the cost structure of the task. Our work differs in that we seek for a method to provide, for each task, review authors with a description of price vs error trade-off, an optimal choice of parameters for a given price, and a set of crowdsourcing strategies that aim at minimizing error estimates. In other words, we don't "simply" aim at classifying papers given the votes from the crowd, but we identify the strategies to obtain such votes by considering the price vs error trade-offs.

Model and Assumptions

Task Model

We consider a crowdsourcing task model that includes set of candidate papers $CP = \{p_1, p_2, ..., p_n\}$ and a textual definition of the scope of the review S. The task is performed by workers in a pool of contributors. In practice this pool is very large and for our purposes we assume it is infinite. We then ask each worker j to label one or more candidate papers as *in* (the paper is in scope or we do not have sufficient evidence to exclude it from the abstract and title) or *out*, based on S. In case of exclusion, workers are asked to provide reason to do so. Figure 1 shows an example task for a review we recently completed.

The result of a task execution is a set of votes $L = \{l_{pj}\}$ representing the binary vote of worker j on paper p. Given the set of votes, we use a classification function $cls(L)$ that takes the individual votes and aggregates them to derive the in/out decision for each paper.

Finally, we define the costs (*loss*) for each error: a cost for a false exclusion $Cost_{fe}$ (deciding that a paper is out when it should have been in), and for a false inclusion $Cost_{fi}$. For simplicity we model them as a cost ratio cr that defines how more costly a false exclusion is with respect to a false inclusion. False exclusions are typically more costly since we may be missing an important paper, while a false inclusion "simply" means that experts will need to go over that paper again. The value of cr depends on the subjective opinion of review authors.

A run of a crowdsourcing task proceeds as follows[4]: first each worker is shown a set N_t of test questions with known labels. If the worker answers them correctly, they move to the work phase, where they can provide useful votes (that is, label unknown candidate papers). Even during the work phase, test questions may be injected with a defined frequency and the contributor is considered trusted (their results are not discarded) only if they keep answering the test questions correctly. The run continues until a given number of labels per paper J has been reached. We assume test questions are created by authors based on the problem at hand, that is, based on a screening they perform over a handful of papers. Typically, creating ten test items that include inclusion and exclusion examples is sufficient for crowdsourcing purposes.

In the simplest case a task will have just one run, but we can envision that a run may leave us with uncertainty over some papers and we may want to have additional runs focused on uncertain papers.

Last but not least, each task has a *price*. The price tag is affected by: i) the unit cost per label (how much we pay workers for labeling a paper or an exclusion criterion), ii) the total

[3]http://www.stat.berkeley.edu/ mossel/teach/ SocialChoiceNetworks10/ScribeAug31.pdf

[4]The choice of the model is also guided by what we can do today with platforms such as Crowdflower

number of votes asked, and iii) the number of test questions. The first two are rather obvious, while the third requires an explanation: with infinite workers, to get accurate results we might simply have a very large number of test questions so that we are sure that only trusted, competent workers remain in the task. In many systems test questions are not paid, so this costs zero. In practice this is not possible: the ethics of this are questionable at best, non-cheaters would do a lot of unpaid work, and we, as task providers, would get bad ratings, impacting our future ability to crowdsource. In this paper we take this into account by increasing the price per judgment by a factor $\frac{N_l+N_t}{N_l}$, where N_t is the number of initial test questions and N_l is the number of valid judgments that a worker gives on non-test papers (i.e., the number of votes from a worker who remains above the threshold *tr*). This essentially states that people who pass the test are in fact paid also for test questions. As N_l grows our factor becomes ineffective and others can be chosen, but in our case N_l is small[5]. Alternative models can be derived, also including a penalty for high test frequencies, but for presenting the concepts and ideas of this paper this is sufficient. The classification cost for a paper is therefore expressed as follows, where US is the cost per vote and $\frac{N_l+N_t}{N_l}$ is the corrective factor:

$$PPP = UC \cdot J \cdot \frac{N_l + N_t}{N_l} \quad (1)$$

In the end we want to perform candidate paper selection with high accuracy (minimizing the loss) and minimal price. A specific point of interest lies in whether the crowd can achieve an accuracy similar to (or better than) that of experts at a comparable cost, while ensuring transparency and impartiality of the whole process.

Probabilistic Model

To reason about the model and identify strategies and parameters we define a probabilistic model that describes the characteristics of i) tasks and ii) workers. Both come into play to identify the optimal crowdsourcing strategy and to set the crowdsourcing parameters.

With respect to the task, we model the following:

1. Our *belief* on the proportion of candidate papers that should be included. This is important because it affects the classification function. We do not assume that authors necessarily have such a prior belief, and we discuss later how this parameter can be set or estimated.
2. The *difficulty* level of each paper: we need to account for the fact that not all candidate papers are equal, meaning that some papers may be harder than others to classify. In this paper we model difficulty with a uniform distribution (which we can parametrize with a variety of priors, such as the commonly used $Beta(\alpha, \beta)$ or priors as suggested in (Whitehill et al. 2009)).

[5]In practice, depending on the task settings, it may not always easy to enable a worker to label many papers due to the fact that many concurrent workers access the task in parallel and the available work finishes very quickly.

With respect to the workers, we assume that in the worst case workers answer randomly, which means a 0.5 probability of a correct label. The proportion of cheaters is modeled by a Bernoulli random variable Z. For non-cheaters, we initially assume a uniform accuracy from 0.5 to 1. The accuracy probability function is therefore a mixture of a point mass at 0.5 and a density in the (0.5, 1) range.

$$pdf(a) = z \cdot \delta(a - 0.5) + 2 \cdot (1 - z) \quad (2)$$

for $0.5 \leq a < 1$. In the function, δ is the impulse function, while the uniform density is multiplied by 2 (as it is in the $(0.5, 1)$ interval only) and by $(1 - z)$ as the density applies only to non-cheaters. In this paper we do not include more complex cases that include a confusion matrix or priors on the initial probability, but the concepts can be extended to that case.

Calculating error cost and price

Now that we have a model we can reason about *strategies* for crowdsourcing literature reviews and assess them based on *assumptions* we can make related to the model.

The goal is to i) identify which aspects of the model impact the selection of strategies and results, ii) estimate the model parameters (or at least refine our prior, when available) based on actual experimental data, and iii) derive which strategies can lead to good results in terms of error cost (loss) and price. Because each problem is different (and even varies also depending on how we title or present the task, as discussed later), the statistical parameters will also vary, so while we can inform our priors via experiments, each task may have to refine the estimation on the go.

We begin by studying a simple version of the model and a simple crowdsourcing *strategy*. In general, a crowdsourcing task for literature review can be comprised of a number of *runs*, where in each run k we submit a subset CP_k of the candidate papers CP to the crowd, collecting a given number J^k of labels per paper. Furthermore, we start each run with a belief B^k on the proportion of papers to be included, if available (and initially assumed to be 0.5 if there is no estimate). A run R^k is therefore defined by a tuple (CP^k, N_t^k, J^k, B^k).

In the simplest strategy the task consists of one run where we submit all papers and seek for J votes per paper. A classification function will then classify the paper based on the cost ratio cr, trying to minimize the loss while fitting within an experiment budget.

The objective for the algorithm here, before even proceeding with the classification, is to i) estimate the optimal values for task parameters that we (as managers of the crowdsourcing process) can play with, such as number of test questions N_t, the requested judgments per paper J, and the classification function, and ii) provide the scientists with a *budget vs expected loss* curve, showing the error cost depending on the budget, assuming that for each budget we choose the best (lowest loss) configuration identified. The only input explicitly required by the authors is the cost ratio, which is subjective.

The expected error cost (loss) for each paper is given by formula 3, where P(FE) and P(FI) denote the probability of false exclusion and false inclusion.

$$Loss = cr \cdot P(FE) + P(FI) \quad (3)$$

Considering that we obtain J judgments (votes) per paper, if we decide to exclude a paper after we obtain J_t exclusion votes or more for such a paper, the probability of a false exclusion is given by equation 4, where θ_i is the (initially unknown) probability that the correct decision for a paper is inclusion, and $\overline{a}_s$ represent the expected accuracy of workers who pass the test phase. The formulas descend from the observation that P(FE) = $P(decision = exclude/correct = include) \cdot P(correct = include)$, and vice versa for P(FI).

$$P(FE) = \theta_i \cdot \sum_{J_t \leq k \leq J} \binom{J}{k} \cdot (1-\overline{a}_s)^k \cdot \overline{a}_s^{J-k} \quad (4)$$

$$P(FI) = (1-\theta_i) \cdot \sum_{J-J_t < k \leq J} \binom{J}{k} (1-\overline{a}_s)^k \overline{a}_s^{J-k} \quad (5)$$

In this formula, θ_i is an unknown parameter we need to estimate, $\overline{a}_s$ is also an unknown parameter on which, however, we can have some control by adjusting the test questions N_t to filter inaccurate workers, while J and J_t can be set to optimize loss.

The accuracy $\overline{a}_s$ of the population that survives N_t tests is distributed as follows: if we denote with z_s the proportion of cheaters in the population that survives the test, which can be derived from Bayes ($z_s = P(test\_passed/cheater) * P(cheater)/P(test\_passed)$), then

$$Z_s = \frac{z \cdot 0.5^{N_t}}{(z \cdot 0.5^{N_t}) + (1-z)\frac{2}{N_t+1} \cdot (1 - \frac{1}{2^{N_t}+1})} \quad (6)$$

Consequently, by using again Bayes for deriving how the accuracy of non cheaters, initially uniform, is reshaped by the test questions, we obtain:

$$f^t(a) = z_s \cdot \delta(a-0.5) + (1-z_s)\frac{2^{(N_t+1)} \cdot (N_t+1)}{2^{N_t+1}-1} \cdot a^{N_t} \quad (7)$$

for $0.5 \leq a < 1$

The expected accuracy $\overline{a}_s$ of this population is $E[x] = \int_{0.5}^{1} x \cdot f^t(x)dx$ and is therefore shown in Equation 8.

$$\overline{a}_s = \frac{z_s}{2} + (1-z_s)\frac{2^{(N_t+1)} \cdot (N_t+1)}{2^{N_t+1}-1}\left(\frac{1-(0.5)^{N_t+2}}{N_t+2}\right) \quad (8)$$

Error minimization and error/price tradeoffs

We begin our discussion on algorithms by assuming a single-run strategy.

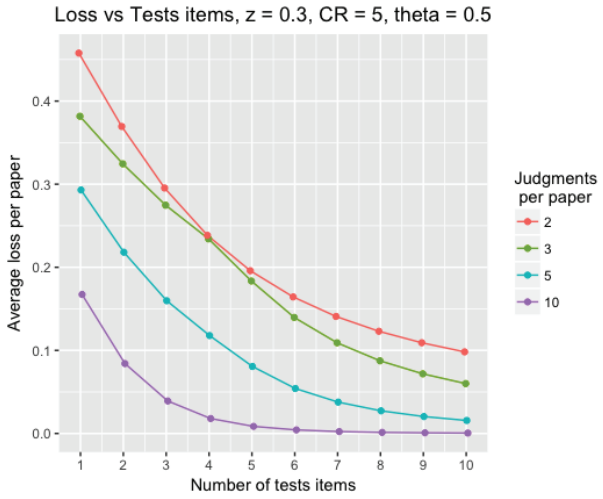

Figure 2: Expected loss depending on the number of test questions and of judgments per paper (real $\theta_i = 0.5$).

Single-run strategy with simple majority voting.

In this approach we simply classify papers using majority voting, which is the approach most commonly supported by crowdsourcing platforms. For each combination of N_t, N_l and J we can compute the total price tag of the experiment as well as estimate the loss via equation 3, where J_t is set to $J/2$ (rounded to the upper integer), as shown in Figure 2. As we have no knowledge of θ_i, we assume a value (such as 0.5, though different values can be set if the task requester has a prior belief). In practice, values of N_t and J over 10 result in near-zero error cost, so computing loss for higher values can be easily done but is rarely needed.

The result can be plotted as done in Figure 3. The decision of the optimal price/loss point is left to the user as it depends on subjective considerations as well as available budget. Each budget corresponds to an optimal choice of N_t, N_l and J that fits in the budget with minimal loss, so that once we have the requestor's decision we can configure the crowdsourcing task. Notice that for now we are assuming that our initial guess of θ_i is correct.

We can then classify the paper using majority voting. Figure 4 shows the performance of this algorithm (denoted by MV in the legend) in terms of expected loss, assuming an initial run with five tests. Figures 4(b), top and bottom, differ from Figures 4(a) as we assume a more difficult set of papers, in this case simulated by scaling down the accuracy of non-cheaters to the 0.5-0.7 range. Furthermore, the top charts have a lower values for J and cr. The increase with θ_i here is due to the fact that this method does not consider the cost ratio which typically penalizes false exclusions. Therefore, error cost grows with the proportion of included papers.

Single-run strategy considering cost ratio.

The obvious improvement to the baseline is to consider the cost ratio. This time, for each value of N_t and J we can minimize the loss (according to equation 3) by selecting the

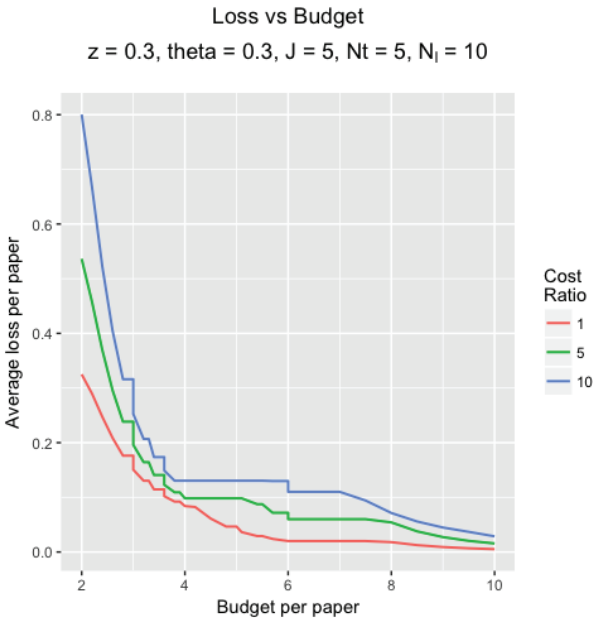

Figure 3: Expected loss that can be achieved depending on the budget (real $\theta_i = 0.3$).

optimal value for J_t. Again, here we only "guess" a θ_i or set it based on the requester belief (in the following chart we assume an initial belief of 0.5). The minimization can be done using classical minimization algorithms (Arora 2015) but also by computing the values given that we have a small number of discrete variables. For each combination we have again a price point and we can plot again loss vs price chart, ask the user to point to an acceptable compromise, determine the parameters and run the task as for the previous case.

As we can see from the results in Figure 4(a) (the label for this algorithm is SCR), this algorithm performs better for high values of θ_i. For $\theta_i = 0.5$ all algorithms behave similarly as the initial assumption of $\theta_i = 0.5$ holds, while for low value of θ_i the loss is higher. This is because we tend to err on the side of inclusion, so for low values of θ_i we get higher errors. However for difficult papers where the accuracy is very low, the error actually grows with θ_i, because the probability of false exclusion goes up and if workers are not precise and we do many errors, we pay a price which is not compensated by erring on the side of inclusion.

Single-run strategy with basic parameters estimation.

The value of the parameters θ_i and z_s plays a role in the loss function, and the cost ratio is also important for determining the optimal classification function given the outcome of a run (in our case, for determining J_t which is the only parameter left to play with once we have concluded a run). Therefore, we assume we can improve on the above method by estimating θ_i. There are many ways in which this can be done. One option is to again use *majority voting* but only for performing an initial classification. Based on this, we compute the proportion of included papers and take this as an estimate for θ_i, more informed than an initial guess of 0.5. We then compute the accuracy of each worker (as percentage of "correct" answer based on majority voting classification), and with estimates of θ_i and $\overline{a}_s$, we then compute the optimal value for J_t based on equation 3, and correspondingly we know the minimal loss we can achieve for each price.

As we can see from the results in Fig 4(a) (the label for this algorithm is BPE), this algorithm performs significantly better than the previous ones for all values of θ_i except 0.5 (where the guess of the simpler algorithms is correct).

Single-run strategy with EM-based parameters estimation.

We can improve on the above algorithm by iterating over estimates of the parameters until convergence. A common method for doing so is to leverage Expectation Maximization (Dempster, Laird, and Rubin 1977; Dawid and Skene 1979). In our model, the data is presented as a Bayesian network, where there are two types of variables: 1) the observable votes provided by workers, and 2) hidden variables, such as θ_i, the workers' accuracy, and the classification for each paper. Via the EM algorithm we computes the correctness of values given the accuracies of the workers that support it. See (Jeff Pasternack 2011) for the details and examples of EM-based for data aggregation. The results shown in Figure 4 indicates that EM is equal to basic estimation and slightly better when accuracy is low.

Multi-run Strategies.

The big limitation in all of the previous algorithms is that we run the crowdsourcing task "in the dark". We "guess" the value of the parameters and, based on this, set the number of tests and of judgments, leaving the optimization to the post-task analysis phase, when the money has been already spent. We can improve on this by running a small test-run whose purpose is to obtain initial estimates for θ_i and z_s. Once we get initial estimates, we can compute and plot again the budget vs loss chart, and based on the estimates and within the confines of the budget, minimize the loss, but this time with the ability to modify N_t and J based on the estimates. We call this a *horizontal multi-run strategy* as we cut the list of papers horizontally. The approach assumes that the initial sample of papers is representative of the whole set, and in absence of specific knowledge this means that we randomly reshuffle the papers before selecting the initial P papers for the estimation run.

The results are shown in Figure 5, depicting the price per paper we can obtain with a multirun strategy that has the same loss of a single run strategy, run with a budget of 7.5, and optimized with EM-based algorithm. We can see that for values of θ_i close to 0 or 1 a multirun strategy obtains savings of approximately 20%.

Multi-run strategies are particularly important when the difficulty of the task is unclear: the difficulty affects the accuracy as pointed out in (Whitehill et al. 2009), so that papers in certain areas may get lower accuracy than others. Similarly, we can apply a *vertical* multi-run strategy where

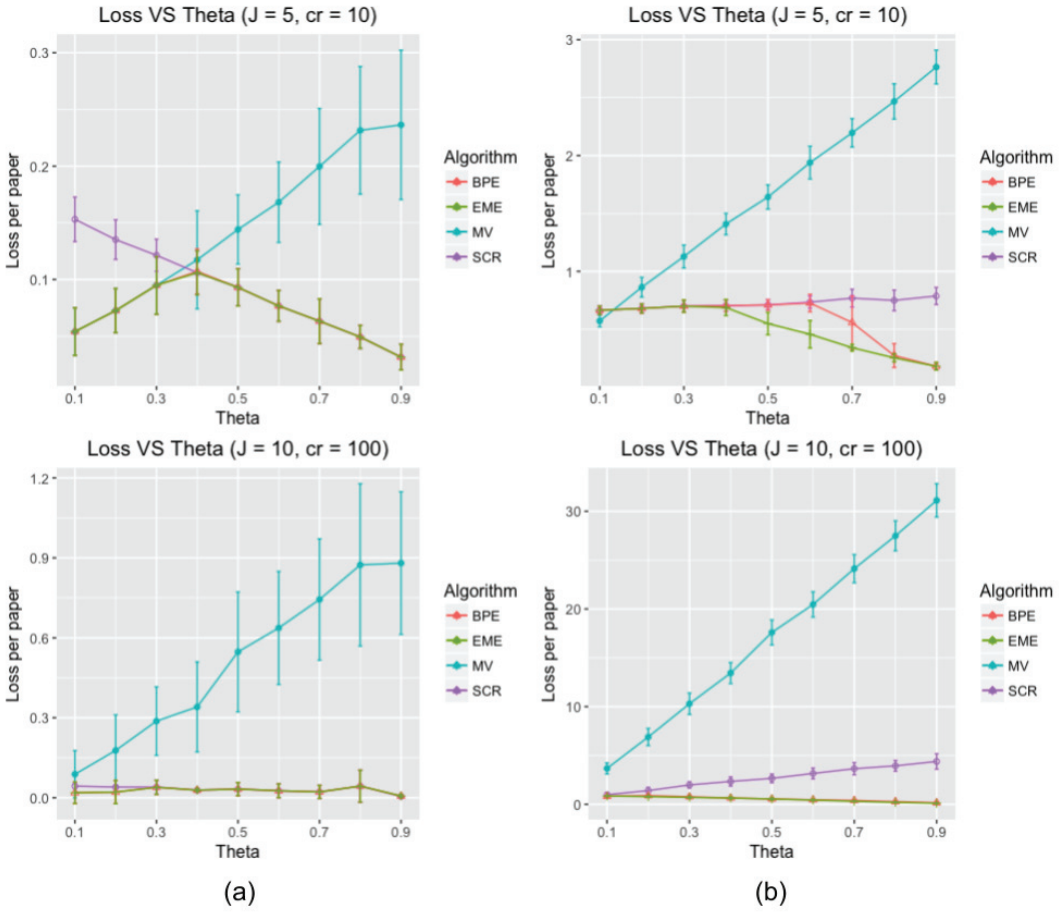

Figure 4: Expected loss for each algorithm. With no difficulty bias (a) and with difficulty bias reducing worker accuracy to a 0.5-0.7 range (b). MV is majority voting, SCR is Simple strategy with cost ratio, BPE is basic parameter estimation, and EME is expectation maximization. The simulation is based on 1000 papers, $N_t = 5, N_l = 10, z = 0.3$.

θ	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
PPP M	6.15	6.15	6.15	6.15	7.5	7.5	7.5	6.15	6.15

Figure 5: Price per paper with a multirun strategy that has the same loss of a single run strategy at a price of 7.5, optimized with EM-based algorithm. Shown for different values of θ_i. Estimation based on $N_t = 5, cr = 10, z = 0.3$, run of 1000 papers.

we collect *one* vote on *all* papers, and use this to estimate the parameters, and proceed with collecting a second vote, and so on. We do not dicuss this further in this paper, but the idea and methods are similar to the horizontal case.

Analysis via Crowdsourcing Experiments

In the winter and spring of 2017 we run a series of experiments on Crowdflower to assess our results and estimate parameters based on actual crowdsourcing scenarios, as well as to understand how such a task can be framed and how sensitive it is to how we word the question or to the difficulty of the papers.

We ran a total of 16 experiments with different settings, asking workers to label a total of 50 papers taken from two systematic reviews, one done by us in an area across computer science and social sciences reviewing technology for fighting loneliness, using fairly common terminology, and the other in medicine (Veronese et al. 2017) having more complex exclusion criteria, with 26 and 24 papers respectively [6]. We collected votes by 2896 respondents (807 of which passed the test phases). The price of each label per paper was also experimented, ranging from 0.22$ to 0.35$, which corresponded to approximately 10 to 15 $/hour. The purpose of the run was not so much to use Crowdflower to get the results, but to understand the workers response in terms of accuracy and speed on real tasks.

The first observation is that the price point is considered acceptable by workers. Overall, the job was rated from 3.3 to 4 in a 5-point scale, and we understand from Crowdflower

[6]The detailed description of all experiments is available at https://tinyurl.com/csexphc

that this is above average. Interestingly, there is a high variance so that sometimes a lower pay resulted in higher rating for two different tasks with the same settings. Classifications based on exclusion criteria generally get higher ratings for the same pay. On average the tasks attracted one worker every 20 seconds. Because of the large pool of workers that end up working concurrently, each worker cannot rate a high number of papers simply because we quickly reach the desired number of votes per paper.

Another observation is that the worker accuracy changes a lot depending on the subject area. The paper in the medical area, which included complex criteria for determining scope or exclusion obtained an average accuracy of 59%, versus 83% for the technology paper. Interestingly, the accuracy depends on the title we give to the task ("classify a text" vs "screen scientific papers"), probably as titles that convey that the task is complex tend to discourage the casual worker, and we know that workers correctly perceive task complexity (Yang et al. 2016). If we word tasks properly and the problem is sufficiently simple, then as shown by Equation 6 the average accuracy after just a few test questions is very high, and classification errors, even using simple majority voting, are low. In this case the classification can be very precise and indeed, in our experiments, in half of the cases (4) where we recorded an "error" from the crowd, the error was on our side meaning that our "gold" data turned out to be not so gold.

We can use the accuracy distributions as derived from Crowdflower and feed them to the algorithms described earlier to compute task settings for relatively easy and relatively hard paper classification problems, and estimate loss for, e.g., a maximum budget of 1$ per paper and a salary of 20cent per answer. For the medical domain case, the optimal algorithm produces N_t=10 and J=2, giving a cost per paper of 80 cents and an expected error loss for $cr = 10$ of 0.15 if $\theta_i = 0.5$. For reviews where the real $\theta_i = 0.1$ the loss is 0.08 for a cost of 1$ per paper (optimal parameters are $N_t = 7$ and $J = 3$). For the ("easier") technology review we can instead reach a loss of 0.11 when the real $\theta_i = 0.5$ (cost of 80cents per paper, $N_t = 10, J = 2$) and for real $\theta_i = 0.1$ the loss is 0.08 for a cost of $N_t = 6, J = 3$, budget of 96 cents per paper.

Notice that 80 cents per paper is a reasonable figure (also obtained with generous assumptions on the time actually spent working on each paper - in practice it is probably possible to achieve lower costs): in our preliminary survey of over 20 authors of recent literature reviews, respondents reported an average of 1.5 person-months of effort spent in this phase. For a typical screening of approximately 1000 papers the price tag is therefore relatively low.

Summary and Limitations

The analysis indicates that crowdsourcing literature reviews can be done with high precision and costs figures that are reasonable with respect to what authors spend today in terms of effort. Different algorithms can be used to identify the parameters of the crowdsourcing task and the best algorithm we identified based on a multi-run strategy significantly outperforms basic EM (with even larger margins when compared with other simpler algorithms). The work has several limitations: in this presentation we could only include a few comparisons and discussions. As the model (and real life scenarios) have many parameters, a more in depth discussion and analysis is needed. Furthermore, a deeper understanding of the accuracy as derived from Crowdflower is needed as it is affected by many "little things" (as we experienced almost by chance) such as the wording of the task title and content, the type of papers, the availability of a rich set of examples for the worker, and so on. Algorithms still have room for improvement, for example in terms of finding the optimal number of papers to consider for the initial run of the multi-run strategy. Furthermore, we have not discussed and analyzed the impact of clarity and of ongoing tests submitted to workers who pass the test phases. A detailed comparison with actual errors performed when experts decide inclusion and exclusion is also needed for a comprehensive evaluation of the approach.

Acknowledgement

This project has received funding from the EU Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 690962.

References

Arora, R. K. 2015. *Optimization: algorithms and applications*. CRC Press.

Dawid, A. P., and Skene, A. M. 1979. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society. Series C Applied Statistics* 28(1).

Dempster, A. P.; Laird, N. M.; and Rubin, D. B. 1977. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* 39(1).

Eickhoff, C., and de Vries, A. P. 2013. Increasing cheat robustness of crowdsourcing tasks. *Information retrieval* 16(2):121–137.

Eveleigh, A.; Jennett, C.; Blandford, A.; Brohan, P.; and Cox, A. L. 2014. Designing for dabblers and deterring drop-outs in citizen science. In *Procs of CHI 2014*. ACM Press.

Frey, B. S., and Jegen, R. 2001. Motivation crowding theory. *Journal of Economic Surveys* 15(5).

Garneau, N. L.; Nuessle, T. M.; Sloan, M. M.; Santorico, S. A.; Coughlin, B. C.; and Hayes, J. E. 2014. Crowdsourcing taste research: genetic and phenotypic predictors of bitter taste perception as a model. *Frontiers in integrative neuroscience* 8:33.

Grant, M., and Booth, A. 2009. A typology of reviews: an analysis of 14 review types and associated methodologies. *Health Info Libr J* 26(2):91–108.

H. Riesch H, C. P. 2014. Citizen science as seen by scientists: Methodological, epistemological and ethical dimensions. *TUGBoat* 23(1).

Haidich, A. 2010. Meta-analysis in medical research. *Hippokratia* 14(Suppl 1):29–37.

Harris, J. D.; Quatman, C. E.; Manring, M.; Siston, R. A.; and Flanigan, D. C. 2014. How to write a systematic review. *The American journal of sports medicine* 42(11):2761–2768.

Henderson, L. K.; Craig, J. C.; Willis, N. S.; Tovey, D.; and Webster, A. C. 2010. How to write a cochrane systematic review. *Nephrology* 15(6):617–624.

Hennon, C. C.; Knapp, K. R.; Schreck III, C. J.; Stevens, S. E.; Kossin, J. P.; Thorne, P. W.; Hennon, P. A.; Kruk, M. C.; Rennie, J.; Gadéa, J.-M.; et al. 2015. Cyclone center: can citizen scientists improve tropical cyclone intensity records? *Bulletin of the American Meteorological Society* 96(4):591–607.

Hirth, M.; Hoßfeld, T.; and Tran-Gia, P. 2011. Cost-optimal validation mechanisms and cheat-detection for crowdsourcing platforms. In *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on*, 316–321. IEEE.

Hirth, M.; Hoßfeld, T.; and Tran-Gia, P. 2013. Analyzing costs and accuracy of validation mechanisms for crowdsourcing platforms. *Mathematical and Computer Modelling* 57(11):2918–2932.

Jeff Pasternack, D. R. 2011. Making better informed trust decisions with generalized fact-finding. *IJCAI International Joint Conference on Artificial Intelligence* 2324–2329.

Karger, D. R.; Oh, S.; and Shah, D. 2011a. Budget-optimal crowdsourcing using low-rank matrix approximations. In *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*, 284–291. IEEE.

Karger, D. R.; Oh, S.; and Shah, D. 2011b. Iterative learning for reliable crowdsourcing systems. In *Advances in neural information processing systems*, 1953–1961.

Khan, K. S.; Kunz, R.; Kleijnen, J.; and Antes, G. 2003. Five steps to conducting a systematic review. *Journal of the Royal Society of Medicine* 96(3):118–121.

Law, E.; Gajos, K. Z.; Wiggins, A.; Gray, M. L.; and Williams, A. 2017. Crowdsourcing as a tool for research: Implications of uncertainty. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, CSCW '17, 1544–1561. New York, NY, USA: ACM.

Lintott, C. J.; Schawinski, K.; Slosar, A.; Land, K.; Bamford, S.; Thomas, D.; Raddick, M. J.; Nichol, R. C.; Szalay, A.; Andreescu, D.; et al. 2008. Galaxy zoo: morphologies derived from visual inspection of galaxies from the sloan digital sky survey. *Monthly Notices of the Royal Astronomical Society* 389(3):1179–1189.

Liu, C., and Wang, Y. M. 2012. Truelabel + confusions: A spectrum of probabilistic models in analyzing multiple ratings. In *Procs of ICML2012*. ICML.

Liu, Q.; Ihler, A. T.; and Steyvers, M. 2013. Scoring workers in crowdsourcing: How many control questions are enough? In *Advances in Neural Information Processing Systems*, 1914–1922.

Moher, D.; Cook, D. J.; Eastwood, S.; Olkin, I.; Rennie, D.; and Stroup, D. F. 1999. Improving the quality of reports of meta-analyses of randomised controlled trials: the {QUOROM} statement. *The Lancet* 354(9193):1896 – 1900.

Moher, D.; Liberati, A.; Tetzlaff, J.; Altman, D. G.; Group, P.; et al. 2009. Preferred reporting items for systematic reviews and meta-analyses: the prisma statement. *PLoS med* 6(7):e1000097.

Ng, L.; Pitt, V.; Huckvale, K.; Clavisi, O.; Turner, T.; Gruen, R.; and Elliott, J. H. 2014. Title and Abstract Screening and Evaluation in Systematic Reviews (TASER): a pilot randomised controlled trial of title and abstract screening by medical students. *Systematic reviews* 3(1):121.

Nguyen, A. T.; Wallace, B. C.; and Lease, M. 2015. Combining Crowd and Expert Labels using Decision Theoretic Active Learning. *Proceedings of the 3rd AAAI Conference on Human Computation (HCOMP)* 120–129.

Sampson, M.; Shojania, K. G.; Garritty, C.; Horsley, T.; Ocampo, M.; and Moher, D. 2008. Systematic reviews can be produced and published faster. *Journal of clinical epidemiology* 61(6):531–536.

Smyth, P.; Fayyad, U.; Burl, M.; Perona, P.; and Baldi, P. 1995. Inferring ground truth from subjective labelling of venus images. *Advances in neural information processing systems* 7:1085–1092.

Steinberg, E.; Greenfield, S.; Wolman, D. M.; Mancher, M.; Graham, R.; et al. 2011. *Clinical practice guidelines we can trust*. National Academies Press.

Sun, Y.; Cheng, P.; Wang, S.; Lyu, H.; Lease, M.; Marshall, I.; and Wallace, B. C. 2016. Crowdsourcing Information Extraction for Biomedical Systematic Reviews. In *4th AAAI Conference on Human Computation and Crowdsourcing (HCOMP): Works-in-Progress Track*. 3 pages. arXiv:1609.01017.

Swanson, A.; Kosmala, M.; Lintott, C.; Simpson, R.; Smith, A.; and Packer, C. 2015. Snapshot serengeti, high-frequency annotated camera trap images of 40 mammalian species in an african savanna. *Scientific data* 2:150026.

Takwoingi, Y.; Hopewell, S.; Tovey, D.; and Sutton, A. J. 2013. A multicomponent decision tool for prioritising the updating of systematic reviews. *Bmj* 7191(December):1–8.

Tinati, R.; Kleek, M. V.; Simperl, E.; Luczak-Roesch, M.; Simpson, R.; and Shadbolt, N. 2015. Designing for citizen data analysis: A cross-sectional case study of a multi-domain citizen science platform. In *Procs of CHI 2015*. ACM Press.

Veronese, N.; Facchini, S.; Stubbs, B.; Luchini, C.; Solmi, M.; Manzato, E.; Sergi, G.; Maggi, S.; Cosco, T.; and Fontana, L. 2017. Weight loss is associated with improvements in cognitive function among overweight and obese people: A systematic review and meta-analysis. *Neuroscience & Biobehavioral Reviews* 72:87–94.

Wallace, B. C.; Dahabreh1, I. J.; Schmid, C. H.; Lau1, J.; and Trikalinos, T. A. 2013. Modernizing the systematic review process to inform comparative effectiveness: tools and methods. *J. Compar. Effect. Res.* 2(3).

Weiss, M. 2016. Crowdsourcing literature reviews in new domains. *Technology Innovation Management Review* 6(2):5–14.

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

Wright, R. W.; Brand, R. A.; Dunn, W.; and Spindler, K. P. 2007. How to write a systematic review. *Clinical orthopaedics and related research* 455:23–29.

Yang, J.; Redi, J.; DeMartini, G.; and Bozzon, A. 2016. Modeling task complexity in crowdsourcing. In *Proceedings of The Fourth AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2016)*, 249–258. AAAI.