# Approximating the Wisdom of the Crowd

**Şeyda Ertekin**[1,3], **Haym Hirsh**[2,3], **Cynthia Rudin**[1,3]
[1]MIT Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA 02142
[2]Department of Computer Science, Rutgers University, Piscataway, NJ 08854
[3]MIT Center for Collective Intelligence, Massachusetts Institute of Technology, Cambridge, MA 02142
{seyda,rudin}@mit.edu, hirsh@cs.rutgers.edu

## Abstract

The problem of "approximating the crowd" is that of estimating the crowd's majority opinion by querying only a subset of it. Algorithms that approximate the crowd can intelligently stretch a limited budget for a crowdsourcing task. We present an algorithm, "CrowdSense," that works in an online fashion to dynamically sample subsets of labelers based on an exploration/exploitation criterion. The algorithm produces a weighted combination of the labelers' votes that approximates the crowd's opinion.

## 1 Introduction

Crowdsourcing systems are a useful way to get a lot of opinions (or "votes") very quickly. However, in cases where each vote is provided at a cost, collecting the crowd's opinion in large scale labeling tasks can be expensive and may not even be attainable under fixed budget constraints. Because of the open nature of crowdsourcing systems, it is not necessarily easy to approximate the crowd on a budget. In particular, the crowd is often comprised of labelers with a range of qualities, motives, and incentives; some labelers are simply better than others. In order to effectively approximate the crowd, we need to determine who are the most trustworthy and representative members of the crowd, in that they can best represent the interests of the crowd majority. This is even more difficult to accomplish when items arrive over time, and it requires our budget to be used both for 1) estimating the majority vote, even before we understand the various qualities of each labeler, and 2) exploring the various labelers until we can estimate their qualities well. There is clearly an exploration/exploitation tradeoff: before we can exploit by using mainly the best labelers, we need to explore to determine who these labelers are, based on their agreement with the crowd.

The main contributions of this work are a modular template for algorithms that approximate the wisdom of the crowd, including the exploitation/exploration choices, and an algorithm, *CrowdSense*, that arises from specific choices within this template. In an online fashion, CrowdSense dynamically samples a subset of labelers, determines whether it has enough votes to make a decision, requests more if the decision is sufficiently uncertain, and iteratively updates the labelers' quality estimates.

## 2 Related Work

The low cost of crowdsourcing labor has increasingly led to the use of resources such as Amazon Mechanical Turk to label data for machine learning purposes, where collecting multiple labels from non-expert annotators can yield results that rival those of experts (e.g., [1]). Although a range of approaches are being developed to manage the varying reliability of crowdsourced labor (see, for example [2, 3, 4]), the most common method for using crowdsourcing to label data is to obtain multiple labels for each item from different labelers and treat the majority label as an items true label. Sheng et al. [5], for example, demonstrated that repeated labeling can be preferable to single labeling in the presence of label noise, especially when the cost of data preprocessing is non-negligible. A number of researchers have explored approaches that learn how much to trust different labelers, typically by comparing each labeler's predictions to the majority-vote prediction of the full set. These approaches often use methods to learn both labeler quality characteristics and latent variables representing the "true" labels of items (e.g., [6, 7, 8, 9, 10]), sometimes in tandem with learning values for other latent variables such as image difficulty [11, 12], classifier parameters [13, 14, 15], or domain-specific information about the labeling task [12].

Our work appears similar to the preceding efforts in that we similarly seek predictions from multiple labelers on a collection of items, and seek to understand how to assign weights to them based on their prediction quality. However, previous work on this topic viewed labelers mainly as a resource to use in order to lower uncertainty about the true labels of the data. In our work, we could always obtain the true labels since they are the majority vote of the crowd. We seek to approximate the correct prediction at lower cost by decreasing the number of labelers used, as opposed to increasing accuracy by turning to additional labelers at additional cost. In other words, usually the classifier is not known and the task is to try to learn it, whereas here the classifier is known and the task is to approximate it. This work further differs from most of the preceding efforts in that they presume that learning takes place after obtaining a collection of data, whereas our method also works in online settings, where it simultaneously processes a stream of arriving data while learning the different quality estimates for the labelers. [5] is one exception, performing active learning by reasoning about the value of seeking additional labels on data given the data obtained thus far. Donmez et al. [16] simultaneously estimate labeler accuracies and train a classifier using labelers' votes to apply active learning to select the next example for labeling. We discuss the approach taken in [16] further in Section 4 as one of the baselines to which we compare our results.

Finally, our work appears superficially similar to what is accomplished by polling processes. Polls obtain the opinions of a small number of people so as to approximate the opinions of a typically much large population. However, polling presumes knowledge of individuals' demographic characteristics, determining how to extrapolate from the views of a small group with certain demographic characteristics to a desired target population with its own demographic characteristics. Our work knows nothing about the demographics of the labelers, and at its core assumes that however closely a labeler matches the overall crowd is a good predictor of whether the labeler will do so in the future.

## 3 CrowdSense

The task of finding a subset of labelers that can accurately represent the wisdom of the crowd requires a measure of how well the labelers in a set agree with the majority vote of the crowd, both individually and as a group. We address this problem by modeling the labelers' quality estimates as a measure of their agreement with the crowd. Let $L = \{l_1, l_2, \ldots, l_M\},\ l_k : x \to \{-1, 1\}$ denote the set of labelers and $\{x_1, x_2, \ldots, x_t, \ldots, x_N\}$ denote the sequence of examples, which arrive one at a time. We define $V_{it} := l_i(x_t)$ as $l_i$'s vote on $x_t$ and $S_t \subset \{1, \ldots, M\}$ as the set of labelers selected to label $x_t$. For each labeler $l_i$, we then define $c_{it}$ as the number of times we have observed a label from $l_i$ so far, and $a_{it}$ as how many of those labels were consistent with that set of labelers:

$$c_{it} := \sum_{\tilde{t}=1}^{t} \mathbb{1}_{[i \in S_{\tilde{t}}]}, \quad a_{it} := \sum_{\tilde{t}=1}^{t} \mathbb{1}_{\left[i \in S_{\tilde{t}}, V_{i\tilde{t}} = V_{S_{\tilde{t}}\tilde{t}}\right]}$$

where $V_{S_t t} = \text{sign}\left(\sum_{i \in S_t} V_{it} Q_{it}\right)$ is the weighted majority vote of the labelers in $S_t$. Labeler $l_i$'s quality estimate is then defined as

$$Q_{it} = \frac{a_{it} + K}{c_{it} + 2K}$$

where $t$ is the number of examples that we have collected labels for and $K$ is a smoothing parameter, yielding a Bayesian shrinkage estimate of the probability that labeler $i$ will agree with the crowd, pulling values down toward $1/2$ when there are not enough data to get a more accurate estimate. This ensures that labelers who have seen fewer examples are not considered more valuable than labelers who have seen more examples and whose performance is more certain.

Pseudocode for the CrowdSense algorithm is given in Figure 1. At the beginning of an online iteration to label a new example, the labeler pool is initialized with three labelers; we select two labelers that have the highest quality estimates $Q_{it}$ and select another one uniformly at random. This initial pool of seed labelers enables the algorithm to maintain a balance between exploitation

```
1. Input: Examples $\{x_1, x_2, \dots, x_N\}$, Labelers $\{l_1, l_2, \dots, l_M\}$, confidence threshold $\varepsilon$,
   smoothing parameter $K$.
2. Define: $L_Q = \{l^{(1)}, \dots, l^{(M)}\}$, labeler id's in descending order of their quality estimates.
3. Initialize: $a_{i1} \leftarrow 0,\ c_{i1} \leftarrow 0$ for $i = 1, \dots, M$.
4. Loop for $t = 1, ..., N$
   (a) Compute quality estimates $Q_{it} = \frac{a_{it}+K}{c_{it}+2K},\ i = 1, \dots, M$. Update $L_Q$.
   (b) $S_t = \{l^{(1)}, l^{(2)}, k\}$, where $k$ is randomly sampled from the set $\{l^{(3)}, \dots l^{(M)}\}$.
   (c) Loop for $j = 3 \dots M,\ j \neq k$
       i. $\text{Score}(S_t) = \sum_{i \in S_t} V_{it} Q_{it},\ \ l_{\text{candidate}} = l^{(j)}$.
       ii. If $\frac{|\text{Score}(S_t)| - Q_{l_{\text{candidate}},t}}{|S_t|+1} < \varepsilon$, then $S_t \leftarrow S_t \cup l_{\text{candidate}}$. Otherwise exit loop to stop
           adding new labelers to $S_t$.
   (d) Get the weighted majority vote of the labelers $V_{S_t t} = \text{sign}\left(\sum_{i \in S_t} V_{it} Q_{it}\right)$
   (e) $\forall i \in S_t$ where $V_{it} = V_{S_t t},\ \ a_{it} \leftarrow a_{it} + 1$
   (f) $\forall i \in S_t,\ \ c_{it} \leftarrow c_{it} + 1$
5. End
```

Figure 1: Pseudocode for the CrowdSense algorithm.

of quality estimates and exploration of the quality of the entire set of labelers. We ask each labeler to vote on the example, and we pay a fixed price per label. The votes obtained from these labelers for this example are then used to generate a *confidence score*, given as

$$\text{Score}(S_t) = \sum_{i \in S_t} V_{it} Q_{it}$$

which represents the weighted majority vote of the labelers. Next, we determine whether we are certain that the sign of $\text{Score}(S_t)$ reflects the crowd's majority vote, and if we are not sure, we repeatedly ask another labeler to vote on this example until we obtain sufficient certainty about the label. To measure how certain we are, we look at the value of $|\text{Score}(S_t)|$ and select the labeler with the highest quality estimate $Q_{it}$ that is not in $S_t$ as a candidate to label this example. We then check whether this labeler could either change the weighted majority vote if his vote were included, or if his vote would bring us into the *regime of uncertainty* where the $\text{Score}(S_t)$ is close to zero, and the vote is approximately a tie. The criteria for adding the candidate labeler to $S_t$ is defined as:

$$\frac{|\text{Score}(S_t)| - Q_{l_{\text{candidate}},t}}{|S_t| + 1} < \varepsilon \tag{1}$$

where $\varepsilon$ controls the level of uncertainty we are willing to permit, $0 < \varepsilon \leq 1$. If (1) is true, the candidate labeler is added to $S_t$ and we get this labeler's vote for $x_t$. We then recompute $\text{Score}_{S_t}$ and follow the same steps for the next-highest-quality candidate from the pool of unselected labelers. If the candidate labeler is not added to $S_t$, we assign the weighted majority vote as the predicted label of this example and proceed to label the next example in the collection.

## 4 Experimental Results

Our experimental evaluation assesses the predictive performance of CrowdSense from two perspectives. First, we compare CrowdSense with several baselines to demonstrate its ability to accurately approximate the crowd's vote. We then present a modular view of CrowdSense, and show the impact of each module on the algorithm's accuracy, which represents the agreement with the straight majority vote of the entire crowd. We conducted experiments on three separate datasets. MovieLens represents a dataset of 137 movies given ratings by 11 people, where the goal is to find the majority vote of these reviewers. ChemIR is a dataset of 1165 patents, where 11 algorithms from the 2009 TREC Chemistry Track predicted whether a given patent reflects "prior art" for a given new patent. Reuters represents 6904 newswire stories where 13 classification algorithms (e.g., SVM, decision trees, AdaBoost) were trained on the money-fx category and where the goal is to predict the majority prediction of these algorithms. For MovieLens we added 50% and for ChemIR we added 60% noise to 5 of the labelers to introduce a greater diversity of judgements, since all the original labelers had comparable qualities and didn't strongly reflect the characteristics of the problem that we address.

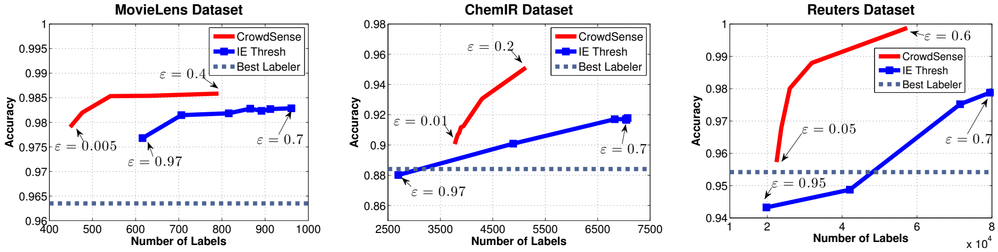

Figure 2: Tradeoff curves. The x-axis is the total number of votes (the total cost) used by the algorithm to label the entire dataset. The y-axis indicates the accuracy on the full dataset.

All reported results are averages of 100 runs with random ordering of examples to prevent bias due to the order in which examples are presented.

We first compared CrowdSense with several baselines: (a) the accuracy of the *average* labeler, represented as the mean accuracy of the individual labelers, (b) the accuracy of the overall best labeler in hindsight, and (c) the algorithm that selects just over half the labelers (*i.e.* $\lceil 11/2 \rceil = 6$ for ChemIR and MovieLens, $\lceil 13/2 \rceil = 7$ for Reuters) uniformly at random, which combines the votes of labelers with no quality assessment. We also compared CrowdSense with IEThresh [16]. IEThresh estimates an upper confidence interval UI for the probability that a labeler will agree with the majority vote and selects all labelers with $UI > \varepsilon \times UI_{\max}$. The $\varepsilon$ parameter in both CrowdSense and IEThresh tunes the size of the subset of labelers selected to vote, so we report results for a range of $\varepsilon$ values. Note that increasing $\varepsilon$ relaxes CrowdSense's selection criteria to ask for votes from more labelers, whereas it causes IEThresh to have a more strict selection policy.

Figure 2 indicates that, for the same fixed total cost, across different values of $\varepsilon$ CrowdSense consistently achieved the highest accuracy against the baselines, indicating that CrowdSense uses a fixed budget more effectively than IEThresh. The other baselines did not achieve the same level of performance as CrowdSense and IEThresh. The accuracy of the best labeler in hindsight (baseline (b)) is indicated as a straight line on the subplots in Figure 2. Baselines (a) and (c), which are the average labeler and the unweighted random labelers, achieved performance beneath that of the best labeler. For the MovieLens dataset, the values for these baselines are 74.05% and 83.69% respectively; for ChemIR these values were 68.71% and 73.13% and for Reuters, the values are 84.84% and 95.25%.

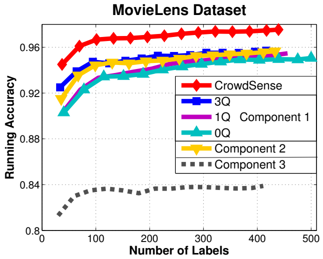

Figure 3: Effect of the modules on CrowdSense's performance.

The algorithm template underlying CrowdSense has three components that can be instantiated in different ways: (1) the composition of the initial seed set of labelers (step 4(b) in the pseudocode), (2) how subsequent labelers are added to the set (step 4(c)), and (3) the weighting scheme that affects the selection of the initial labeler set, the way how the additional labelers are incorporated as well as combining the votes of individual labelers (steps 4(b)(c)(d)). We tested the effect of first component by running separate experiments that initialize the labeler set with three (3Q), one (1Q) and no (0Q) labelers that have the highest quality estimates, where for the latter two additional labelers are selected at random to complete the set of three initial labelers. 3Q removes the exploration capability of the initial set whereas the latter two make limited use of the quality estimates. As seen in Figure 3, all three variants have lower predictive performance compared to CrowdSense. Next, we experimented with the second component by adding labelers randomly rather than in order of their qualities. In this case, exploitation is limited, and the algorithm again tends not to perform as well. To test the effect of the weighting scheme in the third component, we removed the use of weights from the algorithm. This approach performs dramatically worse than the rest of the variants, demonstrating the significance of using quality estimates for labeler selection and the calculation of weighted vote.

## Acknowledgments

We would like to thank to Thomas W. Malone for helpful discussions. Support for this project is provided by MIT Intelligence Initiative ($I^2$).

## References

[1] R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng, "Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2008, pp. 254–263.

[2] E. Law and L. von Ahn, *Human Computation*, ser. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2011.

[3] A. J. Quinn and B. B. Bederson, "Human computation: a survey and taxonomy of a growing field," in *Proceedings of the 2011 Annual Conference on Human Factors in Computing Systems (CHI)*, 2011, pp. 1403–1412.

[4] C. Callison-Burch and M. Dredze, "Creating speech and language data with amazon's mechanical turk," in *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, 2010, pp. 1–12.

[5] V. S. Sheng, F. Provost, and P. G. Ipeirotis, "Get another label? improving data quality and data mining using multiple, noisy labelers," in *Proceeding of the 14$^{th}$ International Conference on Knowledge Discovery and Data Mining (KDD)*, 2008, pp. 614–622.

[6] A. P. Dawid and A. M. Skene, "Maximum likelihood estimation of observer error-rates using the em algorithm," *Applied Statistics*, vol. 28, no. 1, pp. 20–28, 1979.

[7] P. Smyth, U. M. Fayyad, M. C. Burl, P. Perona, and P. Baldi, "Inferring ground truth from subjective labelling of venus images." in *Advances in Neural Information Processing Systems (NIPS)*, 1994, pp. 1085–1092.

[8] P. Smyth, M. C. Burl, U. M. Fayyad, and P. Perona, "Knowledge discovery in large image databases: Dealing with uncertainties in ground truth," in *KDD Workshop*, 1994, pp. 109–120.

[9] G. Kasneci, J. V. Gael, D. Stern, and T. Graepel, "Cobayes: bayesian knowledge corroboration with assessors of unknown areas of expertise," in *Proceedings of the 4$^{th}$ ACM International Conference on Web Search and Data Mining (WSDM)*, 2011, pp. 465–474.

[10] S. K. Warfield, K. H. Zou, and W. M. Wells, "Simultaneous truth and performance level estimation (staple): an algorithm for the validation of image segmentation." *IEEE Transactions on Medical Imaging (TMI)*, vol. 23, no. 7, pp. 903–21, 2004.

[11] J. Whitehill, P. Ruvolo, T. fan Wu, J. Bergsma, and J. Movellan, "Whose vote should count more: Optimal integration of labels from labelers of unknown expertise," in *Advances in Neural Information Processing Systems (NIPS)*, 2009, pp. 2035–2043.

[12] P. Welinder, S. Branson, S. Belongie, and P. Perona, "The multidimensional wisdom of crowds," in *Advances in Neural Information Processing Systems (NIPS)*, 2010.

[13] O. Dekel and O. Shamir, "Good learners for evil teachers," in *Proceedings of the 26$^{th}$ Annual International Conference on Machine Learning (ICML)*, 2009.

[14] Y. Yan, R. Rosales, G. Fung, M. W. Schmidt, G. H. Valadez, L. Bogoni, L. Moy, and J. G. Dy, "Modeling annotator expertise: Learning when everybody knows a bit of something," *Journal of Machine Learning Research - Proceedings Track (JMLR)*, vol. 9, pp. 932–939, 2010.

[15] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *Journal of Machine Learning Research (JMLR)*, vol. 11, pp. 1297–1322, April 2010.

[16] P. Donmez, J. G. Carbonell, and J. Schneider, "Efficiently learning the accuracy of labeling sources for selective sampling," in *Proceedings of the 15th International Conference on Knowledge Discovery and Data Mining (KDD)*, 2009, pp. 259–268.