Semi-Supervised Consensus Labeling for Crowdsourcing

Wei Tang
Department of Computer Science
The University of Texas at Austin
wtang@cs.utexas.edu

Matthew Lease
School of Information
The University of Texas at Austin
ml@ischool.utexas.edu

ABSTRACT

Because individual crowd workers often exhibit high variance in annotation accuracy, we often ask multiple crowd workers to label each example to infer a single *consensus* label. While simple majority vote computes consensus by equally weighting each worker's vote, weighted voting assigns greater weight to more accurate workers, where accuracy is estimated by inner-annotator agreement (unsupervised) and/or agreement with known expert labels (supervised). In this paper, we investigate the annotation cost vs. consensus accuracy benefit from increasing the amount of expert supervision. To maximize benefit from supervision, we propose a semi-supervised approach which infers consensus labels using both labeled and unlabeled examples. We compare our semi-supervised approach with several existing unsupervised and supervised baselines, evaluating on both synthetic data and Amazon Mechanical Turk data. Results show (a) a very modest amount of supervision can provide significant benefit, and (b) consensus accuracy from full supervision with a large amount of labeled data is matched by our semi-supervised approach with much less supervision.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]

General Terms

Algorithms, Design, Experimentation, Performance

Keywords

Crowdsourcing, semi-supervised learning

1. INTRODUCTION

Crowdsourcing has emerged as a major labor pool of exploring human computation for a variety of small tasks over the past few years. Such tasks include image tagging, natural language annotations [14], relevance judging [1], etc. Amazon Mechanical Turk (MTurk) has attracted increasing attention in industrial and academic research as a convenient, inexpensive, and efficient platform for crowdsourcing tasks that are difficult to effectively automate but can be performed by remote workers.

On MTurk, "requesters" typically submit many annotation micro-tasks, and workers choose which tasks to perform. Requesters obtain labels more quickly and affordably, and workers earn a few extra bucks. Unfortunately, accuracy of individual crowd workers has often exhibited high variance in past studies due to factors like poor design or incentives of tasks, ineffective or unengaged workers, or annotation task complexity. Two common methods for quality control are: (a) worker filtering [6] (i.e. identifying poor quality workers and excluding them) and (b) aggregating labels from multiple workers for a given example in order to arrive at a single "consensus" label. In this paper, we focus on the consensus problem; our future work will study a combined approach.

Accurately estimating consensus labels from individual worker labels is challenging. A common approach to this problem is simple Majority Voting (MV) [14, 13, 16], which is easy to use and can often achieve relatively good empirical results depending on the accuracy of workers involved. In MV method, the annotation that receives the maximum number of votes is treated as the final aggregated label, with ties broken randomly. A limitation of MV is that the consensus label for example is estimated *locally*, considering only the labels assigned to that example (without regard to accuracy of the workers involved on other examples).

An alternative is to consider the full set of *global* labels to estimate worker accuracies. These accuracies can then be utilized for weighted voting [9, 8]. A variety of work has investigated means for assessing quality of worker judgments [11] and/or difficulty of annotation tasks [15]. If true ”gold” labels for some examples are first annotated by experts, estimation can be usefully informed by having workers re-annotate these same examples and compare their labels to those of the experts. Snow et al. [14] adopted a fully-supervised Naive Bayes (NB) method to estimate the consensus labels from such gold labels. However, full-supervision can be costly in expert annotation (why we are doing crowdsourcing in the first place). Recent work has studied the effectiveness of supervised vs. unsupervised methods for consensus labeling via simulation [5].

While voluminous amounts of expert data cannot be expected, it may be practical to obtain a limited amount of gold data from experts if there is sufficient benefit to the consensus accuracy we can achieve relative to the expert annotation cost. Similar thinking has driven a large body of work in semi-supervised learning and active learning [12]. In such a scenario, we can estimate consensus labels based

SIGIR 2011 Workshop on Crowdsourcing for Information Retrieval, July 28, 2011, Beijing, China.

This version of the paper (August 22, 2011) corrects errors from the original version of the paper which appeared in the workshop.
.

on whatever information is available to us, labeled and unlabeled examples alike. In this paper, we investigate a semi-supervised approach for consensus labeling. We build upon prior work by Nigam et al. [10] from text classification.

The rest of this paper is organized as follows. §2 describes existing statistical methods for consensus labeling in detail and introduce our semi-supervised approach. §3 introduces the datasets we used in our study. Experimental results based on both synthetic and real MTurk data are reported in §4. We draw conclusions and discuss future work in §5.

2. CONSENSUS LABELING METHODS

A typical crowdsourcing task consists of a set of M examples $E = \{e_m\}_{m=1}^{M}$. Each example is associated with some true label $l(e_m)$ from a set of classes $\{1, \ldots, C\}$. We assume that there are K workers $W = \{w_k\}_{k=1}^{K}$ participating into this annotation task. Each example receives labels from one or more workers. While not common, a given example can actually be annotated multiple times by the same worker (e.g. reusing a "trap question" across multiple worker assignments or validating self-consistency of workers over time by question repetition). Let $n_{mj}^{(k)}$ denote the number of times example e_m receives response j from worker w_k. Let $\{T_{mi}\}$, where $m = 1, \ldots, M$ and $i = 1, \ldots, C$, be the set of indicators for class membership of example e_m such that $T_{mt} = 1$ if t is the true label of example e_m and $T_{mi} = 0$ otherwise.

Majority Vote (MV) assigns the label with the most votes:

$$\hat{l}(e_m) = \arg\max_{c} N(c) \tag{1}$$

as the estimation to the true label for example e_m, where $N(c)$ denotes the number of times example e_m receive response c from all workers.

Expectation Maximization (EM) [3] first estimates the error rates of each worker w_k by a latent *confusion matrix* $[\pi_{ij}^{(k)}]_{C\times C}$, where the ij-th element $\pi_{ij}^{(k)}$ denotes the probability of worker w_k classifying an example to class j given the true label is i, which can be estimated based on each example's class membership as:

$$\hat{\pi}_{ij}^{(k)} = \sum_{m=1}^{M} T_{mi} n_{mj}^{(k)} / \sum_{i=1}^{C}\sum_{m=1}^{M} T_{mi} n_{mj}^{(k)}, \tag{2}$$

and the latent class prior $\{p_i\}_{i=1}^{L}$ is estimated as:

$$\hat{p}_i = \sum_{m=1}^{M} T_{mi}/M. \tag{3}$$

Since the true label for each example e_m is unknown in the unsupervised methods, i.e., T_{mi} is missing, EM uses the mixture of multinomials to describe the quality of workers. Assuming every pair of workers provides independent judgments, the probabilistic model likelihood can be written:

$$\mathcal{L}(p_i, \pi_{ij}^{(k)}) = \prod_{m=1}^{M} \left(\sum_{i=1}^{C} p_i \prod_{k=1}^{K}\prod_{j=1}^{C} (\pi_{ij}^{(k)})^{n_{mj}^{(k)}} \right). \tag{4}$$

Directly estimating the maximum likelihood defined in Equation (4) is difficult since it involves computing product of summation. After we get estimates for latent parameters p_i and $\pi_{ij}^{(k)}$, we can derive new class membership T_{mi} for example e_m such that $T_{ml} = 1$ if l is the estimated true label for example e_m which maximizes:

$$\mathcal{L}(p_i, \pi_{ij}^{(k)}) = \prod_{m=1}^{M} p_i \prod_{k=1}^{K}\prod_{j=1}^{C} (\pi_{ij}^{(k)})^{n_{mj}^{(k)}}. \tag{5}$$

Therefore, using EM algorithm, we can iteratively estimate latent parameters p_i, $\pi_{ij}^{(k)}$ and missing labels T_{mi}, based on Equation (2), (3) and (5), until convergence.

If true labels of examples are all available, the above probabilistic model is reduced to a single multinomial distribution. The likelihood can be simplified as in Equation (5). In this case, Naive Bayes (NB) method can be applied to estimate a more accurate confusion matrix for each worker using Equation (5) with the same assumption that there is no interaction between workers.

What is of great interest in this work is to estimate confusion matrix for each worker when both labeled and unlabeled examples are available for us. In this case, we assume that there is a small set of examples L whose true labels have been provided by domain experts and the set of rest unlabeled examples is denoted as U.

To address this, we propose to a Semi-supervised Naive Bayes (SNB) approach with new likelihood function:

$$\begin{aligned} \mathcal{L}(p_i, \pi_{ij}^{(k)}) = \quad & \prod_{m\in U} \left(\sum_{i=1}^{C} p_i \prod_{k=1}^{K}\prod_{j=1}^{C} (\pi_{ij}^{(k)})^{n_{mj}^{(k)}} \right) + \\ & \prod_{m\in L} p_i \prod_{k=1}^{K}\prod_{j=1}^{C} (\pi_{ij}^{(k)})^{n_{mj}^{(k)}}. \end{aligned} \tag{6}$$

From Equation (6), we can see that the difference between EM and semi-supervised Naive Bayes method is that we have a separate set of examples whose true labels are known a priori. The labeled examples are used to estimated model parameters and then to give "soft" labels for each unlabeled examples. After that, the model parameters are estimated again based on all labels. This procedure continues until convergence. Figure 1 presents SNB pseudocode.

```
Algorithm: Semi-supervised Naive Bayes (SNB)
Input: A set of labels $\{l_{km}\}$ from worker $w_k$ to example $e_m$
and a set of true labels $\{c_l\}$ to some examples $e_l \in E$.
Output: Confusion matrix for $\pi_{ij}^{(k}$ for each worker $w_k$, class
prior distribution $\{p_i\}_{i=1}^{C}$ and estimated consensus label $\hat{T}(e_m)$
for example $e_m$.
Steps:
  1. Initialization: initialize labels of unlabeled examples by
     majority voting over worker judgments;
  2. Loop until there is no further improvement:
     (a) Given the true labels for labeled examples and
         estimated labels for unlabeled examples, estimate
         the latent model parameters $p_i$ and $\pi_{ij}^{(k)}$ using
         Equation (3) and (2), respectively;
     (b) re-estimate consensus labels for unlabeled examples
         using Equation (5);
```

Figure 1: Semi-supervised Naive Bayes algorithm.

3. DATA

This section describes two datasets used to evaluate our methods: a synthetic dataset with labels automatically generated via simulated workers, and a dataset of actual labels collected from MTurk workers. Evaluation using these datasets is described in §4.

3.1 Synthetic Data

To simulate workers with differing accuracy and control for the ratio of labeled vs. unlabeled examples, we generate a synthetic data set for binary classification with 8000 examples (uniformly) randomly assigned to each class. We generate a pool of 800 workers, each with a simple Bernoulli accuracy parameter $p_k \sim \mathbb{U}[0.3, 0.7]$. The number of labels per example is randomly set between 2 to 8, and the assignment of workers to examples is selected uniformly at random (with replacement, though workers annotate each example e_m at most once: $\forall j \in C, n_{mj}^{(k)} <= 1$).

3.2 MTurk Data

To evaluate the effectiveness of our methods on human relevance judgments, we used topical judgments collected via MTurk for the TREC 2010 Relevance Feedback Track [2].

Judging was performed via a mostly pre-existing judging interface described in [4]; Figure 2 gives a screenshot of the judging interface. Workers were provided a NIST TREC[1] *title*, *description*, and *narrative* for each search topic and asked to assess topical relevance of five ClueWeb09[2] Webpages per MTurk Human Intelligence Task (HIT). We offered workers US $0.05 per HIT. Relevance judging was predominantly ternary, with multiple choice responses "very relevant", "relevant", and "not relevant". To protect crowd workers from malicious attack pages, workers judged rendered Webpages in one of three forms: as images, PDFs, or text. To allow for the possibility of processing error, a fourth multiple choice option ("I could not view... the webpage...") allowed workers to report such problems explicitly.

For quality control, one Webpage per HIT either had a prior NIST judgment or was intentionally broken (in which case the correct response was the fourth multiple choice option). In our experiments, 3,275 of 19,033 total topic-document examples had prior expert labels (we exclude broken link examples and judgments in this study). We also collapse "very relevant" and "relevant" categories, yielding binary labels distinguishing relevance vs. non-relevance only.

Figure 3 shows statistics of worker accuracy vs. the number of annotations per worker. Each point represents a worker, the x-axis (in log scale) denotes the number of annotations provided by the given worker, and the y-axis shows worker accuracy relative to prior expert labels. We see that most workers provide a few low quality annotations.

4. EXPERIMENTS

We report a set of experiments performed on both synthetic data and real MTurk data described in §3. Results show that (a) a very modest amount of supervision can provide significant benefit, and (b) consensus accuracy from full supervision with a large amount of labeled data is matched by our semi-supervised approach with much less supervision.

We compare three methods: (1) unsupervised MV and EM baselines; (2) supervised NB trained on labeled examples only; (3) SNB using labeled and unlabeled examples.

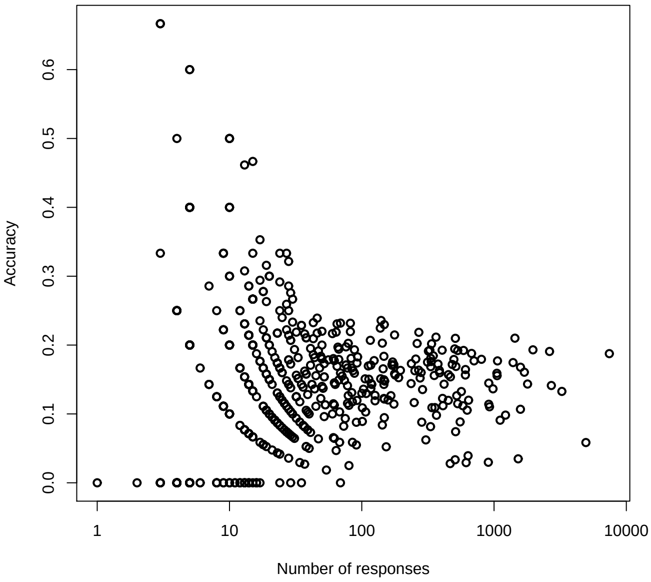


Figure 3: Worker accuracy vs. number of annotations per worker in the MTurk dataset.

4.1 Supervised vs. unsupervised

In our first set of experiments, we compare supervised NB to unsupervised MV and EM. For both synthetic and MTurk datasets, we randomly partition data into train (2048 examples) and test folds (remaining examples). We incrementally vary the number of training examples used for supervision by powers of two (e.g. 128, 256, 512, 1024, 2048); when less than the full training set is used, remaining training data is ignored. We measure accuracy of consensus labels obtained, running each experiment 10 times and averaging for result stability.

Figure 4 and Figure 5 show the learning curve of supervised NB with increasing amount of supervision vs. unsupervised MV and EM baselines on synthetic and MTurk data, respectively. Note that the accuracies of unsupervised MV and EM methods remain unchanged since the unsupervised methods do not utilize any supervision.

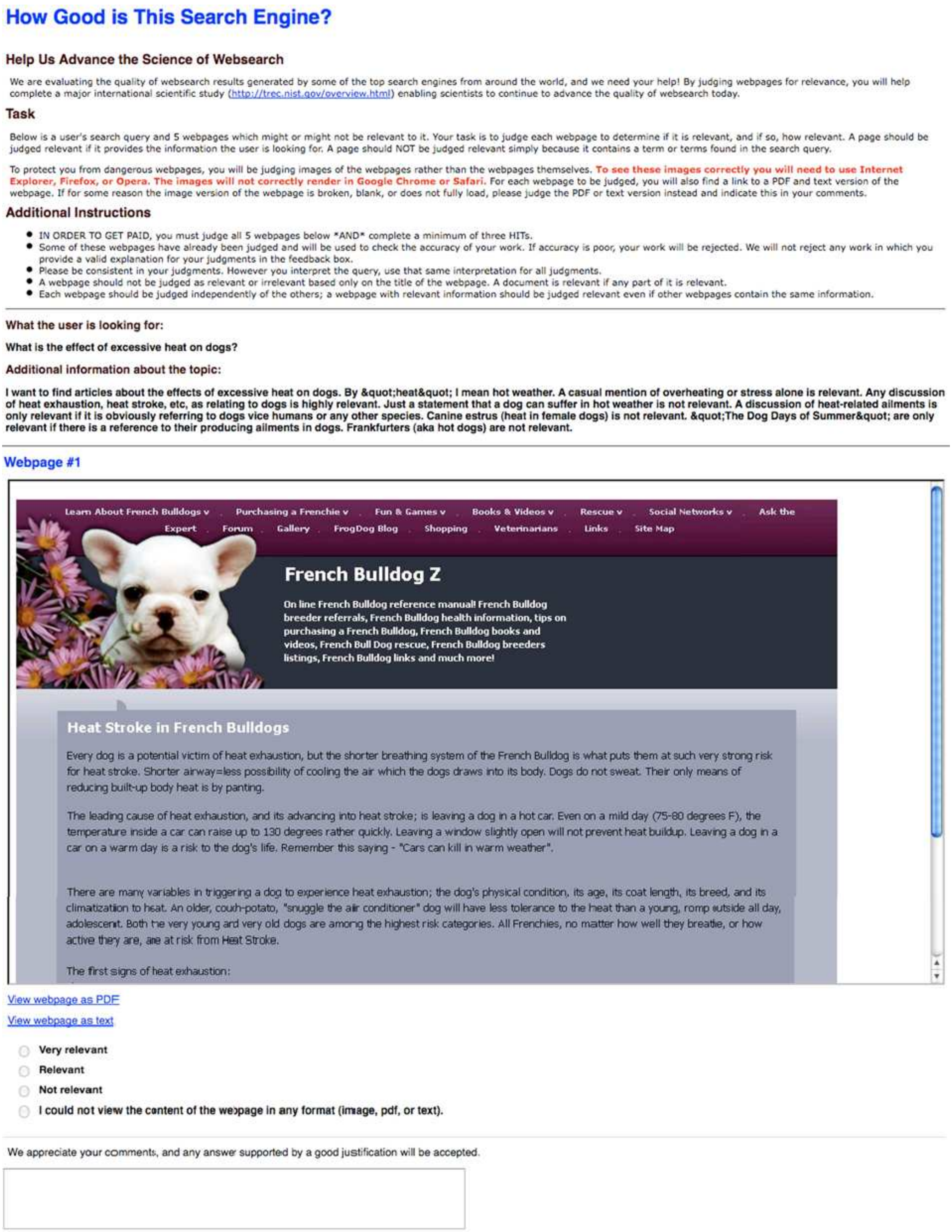
Results for both synthetic and MTurk data are shown and similar. From Figure 4 we can see that, for the synthetic dataset, EM slightly outperforms MV, 75.0% to 74.2%. Both outperform NB when only 128 training examples are used (66.6%). With 512 examples, NB beats EM. When we use the full training set of 2048 examples, NB achieves a far superior 88.7% accuracy, but at a clear cost in expert annotation effort required.

Similarly, from Figure 5, EM also outperforms MV (66.6% to 63.9%) for the MTurk dataset, and NB is once more inferior (62.9%) with only 128 training examples. NB matches EM with 256 training examples, and achieves 70.6% accuracy with the full training set (far less than in the synthetic data, but clearly superior to the unsupervised baselines).

Overall, we see that 256-512 training examples are needed for NB to match or exceed the unsupervised baselines, and that significantly improved accuracy is possible with increasing supervision beyond this.

[1] http://trec.nist.gov

[2] http://lemurproject.org/clueweb09.php



How Good is This Search Engine?

Help Us Advance the Science of Websearch

We are evaluating the quality of websearch results generated by some of the top search engines from around the world, and we need your help! By judging webpages for relevance, you will help complete a major international scientific study (http://trec.nist.gov/overview.html) enabling scientists to continue to advance the quality of websearch today.

Task

Below is a user's search query and 5 webpages which might or might not be relevant to it. Your task is to judge each webpage to determine if it is relevant, and if so, how relevant. A page should be judged relevant if it provides the information the user is looking for. A page should NOT be judged relevant simply because it contains a term or terms found in the search query.

To protect you from dangerous webpages, you will be judging images of the webpages rather than the webpages themselves. **To see these images correctly you will need to use Internet Explorer, Firefox, or Opera. The images will not correctly render in Google Chrome or Safari.** For each webpage to be judged, you will also find a link to a PDF and text version of the webpage. If for some reason the image version of the webpage is broken, blank, or does not fully load, please judge the PDF or text version instead and indicate this in your comments.

Additional Instructions

- IN ORDER TO GET PAID, you must judge all 5 webpages below *AND* complete a minimum of three HITs.
- Some of these webpages have already been judged and will be used to check the accuracy of your work. If accuracy is poor, your work will be rejected. We will not reject any work in which you provide a valid explanation for your judgments in the feedback box.
- Please be consistent in your judgments. However you interpret the query, use that same interpretation for all judgments.
- A webpage should not be judged as relevant or irrelevant based only on the title of the webpage. A document is relevant if any part of it is relevant.
- Each webpage should be judged independently of the others; a webpage with relevant information should be judged relevant even if other webpages contain the same information.

What the user is looking for:

What is the effect of excessive heat on dogs?

Additional information about the topic:

I want to find articles about the effects of excessive heat on dogs. By &quot;heat&quot; I mean hot weather. A casual mention of overheating or stress alone is relevant. Any discussion of heat exhaustion, heat stroke, etc, as relating to dogs is highly relevant. Just a statement that a dog can suffer in hot weather is not relevant. A discussion of heat-related ailments is only relevant if it is obviously referring to dogs vice humans or any other species. Canine estrus (heat in female dogs) is not relevant. &quot;The Dog Days of Summer&quot; are only relevant if there is a reference to their producing ailments in dogs. Frankfurters (aka hot dogs) are not relevant.

Webpage #1

French Bulldog Z

On line French Bulldog reference manual! French Bulldog breeder referrals, French Bulldog health information, tips on purchasing a French Bulldog, French Bulldog books and videos, French Bull Dog rescue, French Bulldog breeders listings, French Bulldog links and much more!

Heat Stroke in French Bulldogs

Every dog is a potential victim of heat exhaustion, but the shorter breathing system of the French Bulldog is what puts them at such very strong risk for heat stroke. Shorter airway=less possibility of cooling the air which the dogs draws into its body. Dogs do not sweat. Their only means of reducing built-up body heat is by panting.

The leading cause of heat exhaustion, and its advancing into heat stroke; is leaving a dog in a hot car. Even on a mild day (75-80 degrees F), the temperature inside a car can raise up to 130 degrees rather quickly. Leaving a window slightly open will not prevent heat buildup. Leaving a dog in a car on a warm day is a risk to the dog's life. Remember this saying - "Cars can kill in warm weather".

There are many variables in triggering a dog to experience heat exhaustion; the dog's physical condition, its age, its coat length, its breed, and its climatization to heat. An older, couch-potato, "snuggle the air conditioner" dog will have less tolerance to the heat than a young, romp outside all day, adolescent. Both the very young and very old dogs are among the highest risk categories. All Frenchies, no matter how well they breathe, or how active they are, are at risk from Heat Stroke.

The first signs of heat exhaustion:

View webpage as PDF

View webpage as text

○ Very relevant

○ Relevant

○ Not relevant

○ I could not view the content of the webpage in any format (image, pdf, or text).

We appreciate your comments, and any answer supported by a good justification will be accepted.

Figure 2: A screenshot of the judging interface for our MTurk task.

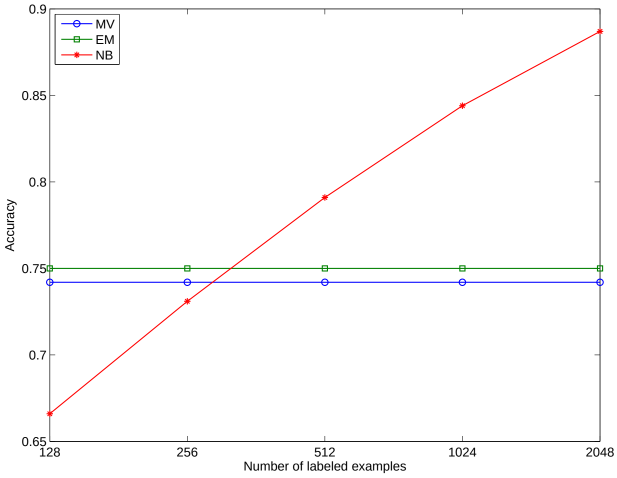


Figure 4: Supervised NB vs. unsupervised MV and EM on the synthetic dataset.

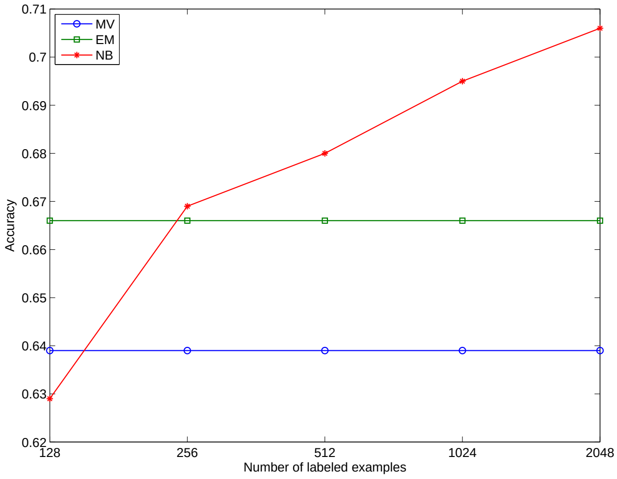


Figure 5: Supervised NB vs. unsupervised MV and EM on the MTurk dataset.

4.2 Semi-supervised vs. supervised

In our second set of experiments, we compare our semi-supervised SNB method vs. supervised NB method, evaluating consensus accuracy achieved across varying amount of labeled vs. unlabeled training data. Starting from each of the same labeled training size values considered in our first set of experiments for supervised NB, we now consider adding additional unlabeled examples in powers of two as before into the training set, though now we have potentially more data to use (up to 5000 unlabeled examples in the synthetic data, and up to 15758 examples with MTurk). As before, we repeat experiments 10 times and average.

Figure 6 and Figure 7 compare semi-supervised SNB method with supervised NB method for synthetic and MTurk data, respectively. Results on both synthetic and MTurk data are quite similar. Each curve in the figures corresponds to a SNB method trained on a different number of (labeled) training examples. The x-axis indicates the number of *additional, unlabeled* examples used for training. While not shown, a value of $x = 0$ (no unlabeled data used) in Figure 6 and Figure 7 would correspond exactly to the accuracy achieved by supervised NB method from Figure 4 and Figure 5, respectively. All curves approach convergence with the full training set (all available labeled and unlabeled data).

Labels for unlabeled examples are automatically estimated by SNB with a given confidence during the training process. Worker labels are then compared to these generated labels and confidence values in order to estimate worker accuracies (in addition to comparing worker labels on expert labeled examples). Figure 4 and Figure 5 intuitively showed that NB consensus accuracy increases with more labeled training data. Figure 6 and Figure 7 reflect this in the relative starting positions of each learning curve of SNB method.

Recall that unsupervised EM method achieved 75.0% consensus accuracy for the synthetic data in Figure 4. From Figure 6 we can see that, with only 256 labeled and 1024

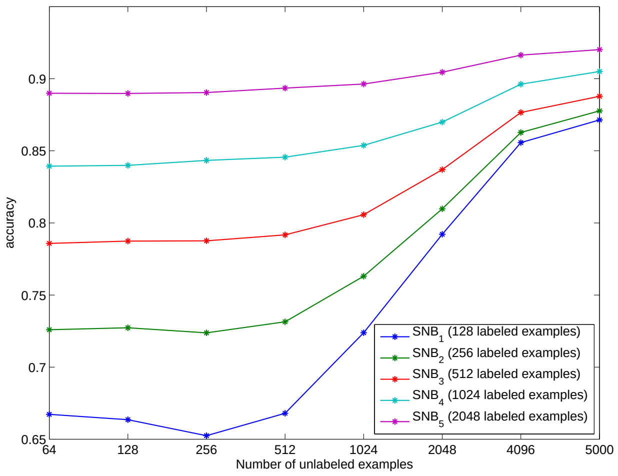


Figure 6: Semi-supervised SNB vs. supervised NB method on the synthetic dataset.

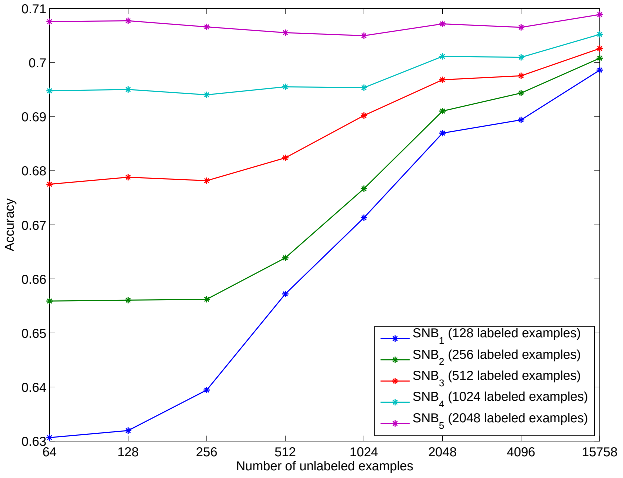


Figure 7: Semi-supervised SNB vs. supervised NB method on the MTurk dataset.

unlabeled training examples, SNB achieve the same performance as unsupervised EM. However, if we have only 128 labeled examples but use all unlabeled examples as training set, SNB achieves *approx* 85% accuracy. At the high end, while NB maxed out with $< 95\%$ with the full training set, SNB achieves $\approx 92\%$.

Similarly, recall that unsupervised EM baseline achieved 66.6% consensus accuracy in Figure 5 for the MTurk data set. From Figure 7, we see that with only 128 labeled and 1024 unlabeled examples as training set, SNB matches EM. With the full set of unlabeled examples for training, however, SNB achieves nearly 70% accuracy and almost the same accuracy that NB achieved when requiring all 2048 labeled examples as training.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a semi-supervised Naive Bayes approach for more accurately inferring consensus labels given relatively less labeled training data for estimating worker accuracy. The proposed method can be used in the situation where we have large amount of unlabeled examples while there is also a small set of expert-labeled examples are available. Experiments on both synthetic and real MTurk data show that (a) a very modest amount of supervision can provide significant benefit, and (b) consensus accuracy from full supervision with a large amount of labeled data is matched by our semi-supervised approach with much less supervision. When expert-labeled examples are limited (e.g. due to time constraints, available budget, or access to personnel), we still can achieve similar consensus accuracy of the fully supervised method via the semi-supervised approach and with large amount of unlabeled examples.

We would like to integrate worker filtering [6] with consensus labeling to better understand how far each can be taken on its own and how to best the two approaches synergistically. We have also only evaluated our consensus methods in the context of one crowdsourcing design and a matching synthetic data setting. Another important direction for quality control is by better addressing other human factors [7, 4]. Better interface or questionnaire design, pricing, or worker recruitment/retention practices, etc. could lessen the degree of filtering/consensus needed, and remaining errors may exhibit different properties. Inversely, less attention to such issues would also present a greater volume and altered distribution of labeling errors for filtering and consensus to correct. Future work should investigate better human factors design and test quality control automation under a wider range of crowdsourcing designs and label noise conditions.

Another interesting direction will utilize predicted labels for unlabeled examples in the annotation process. For example, active learning typically focuses annotation effort on labeling those examples for which current predictions are the most uncertain (and so human labels would be the most informative) [12]. Another direction of work has investigated to what degree providing annotators with predicted labels might reduce time or increase quality of their subsequent labels. Or instead of simply comparing workers' labels with one another's or with expert labels, we might also compare them to our predicted labels based on the current model. This requires a careful balancing act between label informativeness and verifiability: while the most informative labels could not be verified by the model (since they would be too "surprising"), labels which could be trivially verified would not be very informative for model training.

Acknowledgments. We thank Mark Smucker, Catherine Grady, and Chandra Prakash Jethani for their assistance collecting crowdsourced relevance judgments. We also thank the anonymous reviewers for their valuable feedback and suggestions. This work was partially supported by an Amazon Web Services grant and a John P. Commons Fellowship for the second author.

6. REFERENCES

[1] O. Alonso, D. Rose, and B. Stewart. Crowdsourcing for relevance evaluation. *SIGIR Forum*, 42(2):9–15, 2008.

[2] C. Buckley, M. Lease, and M. D. Smucker. Overview of the TREC 2010 Relevance Feedback Track (Notebook). In *The Nineteenth Text Retrieval Conference (TREC 2010) Notebook*, 2010.

[3] A. P. Dawid and A. M. Skene. Maximum likelihood estimation of observer error-rates using the EM algorithm. In *Applied Statistics, Vol. 28, No. 1.*, 1979.

[4] t. . C. Grady, Catherine and Lease, Matthew. In *NAACL-HLT Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 172–179, 2010.

[5] P. G. Ipeirotis. Worker Evaluation in Crowdsourcing: Gold Data or Multiple Workers? `http://behind-the-enemy-lines.blogspot.com/2010/09/worker-evaluation-in-crowdsourcing-gold.html`.

[6] H. J. Jung and M. Lease. Improving Consensus Accuracy via Z-score and Weighted Voting. In *3rd Human Computation Workshop (HCOMP)*, 2011.

[7] G. Kazai, J. Kamps, M. Koolen, and N. Milic-Frayling. Crowdsourcing for Book Search Evaluation: Impact of HIT Design on Comparative System Ranking. In *SIGIR*, 2011.

[8] A. Kumar and M. Lease. Learning to rank from a noisy crowd. In *SIGIR*, 2011. Poster.

[9] A. Kumar and M. Lease. Modeling annotator accuracies for supervised learning. In *WSDM Workshop on Crowdsourcing for Search and Data Mining*, pages 19–22, 2011.

[10] K. Nigam, A. Mccallum, and T. Mitchell. Semi-supervised text classification using EM. In *Semi-Supervised Learning*, pages 33–56. 2006.

[11] V. Raykar, S. Yu, L. Zhao, G. Valadez, C. Florin, L. Bogoni, and L. Moy. Learning from crowds. *Journal of Machine Learning Research*, 11(7):1297–1322, 2010.

[12] B. Settles. Active Learning Literature Survey. Technical Report 1648, University of Wisconsin-Madison Computer Sciences, 2009.

[13] V. S. Sheng, F. Provost, and P. G. Ipeirotis. Get another label? improving data quality and data mining using multiple, noisy labelers. In *KDD*, 2008.

[14] R. Snow, B. O'Connor, D. Jurafsky, and A. Ng. Cheap and fast, but is it good? In *EMNLP*, 2008.

[15] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, and J. Movellan. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *NIPS*, 22:2035–2043, 2009.

[16] H. Yang, A. Mityagin, K. Svore, and S. Markov. Collecting high quality overlapping labels at low cost. In *SIGIR*, pages 459–466, 2010.