Collaborative Active Learning of a Kernel Machine Ensemble for Recognition

Gang Hua[†], Chengjiang Long[†], Ming Yang[‡], Yan Gao[♯]

[†]Stevens Institute of Technology
Hoboken, NJ 07030
{ghua, clong}@stevens.edu

[‡] Facebook
Menlo Park, CA 94026
mingyang2008@u.northwestern.edu

[♯]Northwestern University
Evanston, IL 60208
beargaoyan@gmail.com

Abstract

Active learning is an effective way of engaging users to interactively train models for visual recognition. The vast majority of previous works, if not all of them, focused on active learning with a single human oracle. The problem of active learning with multiple oracles in a collaborative setting has not been well explored. Moreover, most of the previous works assume that the labels provided by the human oracles are noise free, which may often be violated in reality. We present a collaborative computational model for active learning with multiple human oracles. It leads to not only an ensemble kernel machine that is robust to label noises, but also a principled label quality measure to online detect irresponsible labelers. Instead of running independent active learning processes for each individual human oracle, our model captures the inherent correlations among the labelers through shared data among them. Our simulation experiments and experiments with real crowd-sourced noisy labels demonstrated the efficacy of our model.

1. Introduction

Supervised discriminative learning has been one of the main methodologies for advancing research on visual recognition [20, 13]. One of the major difficulties taking such an approach is to collect sufficient trustworthy labeled data for training. To mitigate the heavy workload of labeling, some previous works attempted to train the recognition model with fewer labeled data using semi-supervised learning [10]. Nevertheless, state-of-the-art recognition systems are all based on supervised learning with large quantity of labeled training data [20, 13].

To facilitate more efficient data labeling, some previous works have explored the use of active learning [11, 22, 12, 21, 16, 8], where the learning machine guides the labeler to label the most informative visual examples. However, most previous works on active visual labeling, if not all of them, only investigated the case where a single human oracle is engaged [11, 22, 12, 21, 16, 8]. The problem of active learning with multiple collaborative labelers has largely remained unexplored. Moreover, most previous active learning algorithms assume that labels provided by the human oracle are noise free. Hence, the problem of active learning under the condition that the human oracle may provide somewhat noisy labels is largely neglected.

On the other hand, most of the recent efforts on collecting large scale labeled image datasets, such as ImageNet [5] and LabelMe [19], have exploited crowdsourcing. There are several issues raised when using crowdsourcing systems such as Amazon Mechanical Turk. First of all, there is no active guidance from the system to enable the labelers to more efficiently label the data. Secondly, there is no mechanism to online detect if a labeler is doing the job assignment in the desired way. Last but not least, several studies have shown that the label information collected from Mechanical Turk could be very noisy, either due to irresponsible behaviors from some of the labelers, or due to the inherent ambiguities of the target semantics.

We propose a computational model for collaborative active learning with multiple labelers to address all the above issues, which learns an ensemble kernel machine for classification problems. In our framework, each labeler is running an individual active learning process, where the system naturally guides the labelers to label different images more efficiently towards learning the classifier. These active learning processes are not independent of one another. Our unified discriminative formulation explicitly models the consistencies among all the different active learning processes through the shared data among them. By doing so, not only can we make our active learning model to be more robust to label noises, but also we can derive principled measures to detect irresponsible labelers who are careless about their label quality earlier in the visual labeling process.

There have been some previous works which attempted to use active learning to facilitate crowd-sourced human labeling [23, 1, 22] in various tasks including machine translation [1], named entity extraction and sentiment detection [14], and visual object detection [22]. To handle label noise and irresponsible users, they either perform post-mortem majority voting to reduce label noise [22], or use a pre-labeled gold standard dataset to measure label quality [1], or synchronize labels from different workers on the same examples to conduct online majority vote filtering [14]. None of these seemed to be satisfactory.

Donmez *et al.* [7] proposed a majority voting based con-

fidence interval method to determine the labeling quality of each annotator, which is assumed to be stationary, and used it to select a subset of annotators to query in the active learning process. In their later work [6], a sequential Bayesian estimation method is proposed to deal with non-stationary labeling qualities. Nevertheless, although reliable annotators can be selected, the labels of one data sample from the selected annotators still need to be synchronized, which may not be desirable. Zhao *et al.* [26] proposed an incremental relabeling mechanism which exploits active learning to not only select the unlabeled data to be labeled by the crowds, but also select already labeled data samples to be relabeled until sufficient confidence is built.

Several other previous works have also explored the case of learning models from multiple annotations collected in the absence of gold standard labels. For example, Raykar *et al.* [18, 17] proposed a probabilistic model, which assumes independence of the annotator judgement given the true label. An EM algorithm is developed to alternatively estimate the classification model and measure the performance of the multiple annotators. Dekel and Shamir [3] adapted the formulation of support vector machines (SVMs) to identify low quality or malicious annotators.

However, they assume that each annotator is binary as either good or bad, instead of in a continuous state space. Later, Dekel and Shamir [4] described a method along with its theoretic support for pruning out the low-quality workers by using the model trained from the entire labeled dataset from all workers as ground truth. Chen *et al.* [2] proposed a method to identify good annotators based on spectral clustering in the worker space. The assumption is that good annotators will behave similarly. Yan [24, 25] presented a probabilistic multi-labeler model to learn from the crowds, where the quality of each labeler is modeled by a logistic regression function.

These works provide various insights on how to deal with label noises and malicious labelers. Nevertheless, none of them explored to actively learn an ensemble classifier from multiple noisy labelers. Previous study has demonstrated that an ensemble classifier or multiple classifier system, such as those using bagging, tend to be more resilient to label noises, which partly motivated us to design such a collaborative active learning algorithm to learn an ensemble kernel machine.

We apply the proposed collaborative active learning framework for training classifiers for visual recognition. We validate its efficacy with both simulation experiments and experiments with real crowd-sourced noisy labels from Amazon Mechanical Turk. Our extensive empirical evaluations clearly show that our collaborative active learning algorithm is more robust to label noises when compared with multiple independent active learners, and the learned ensemble kernel classifier can often generalize better to new data. We also show that conducting collaborative active learning naturally leads to more efficient labeling than random learning (i.e., randomly select the next image for a labeler to label). When there are irresponsible labelers, our experiments also manifested that the measure we derived from our model show a very strong signal to detect these irresponsible labelers earlier in the active learning process, which is desired as we want to ban them as early as possible.

Our main contributions are hence four-fold: (1) we propose a unified and distributed discriminative learning model for collaborative active learning among a set of labelers to induce an ensemble kernel machine classifier. (2) From our proposed computational model, we are able to derive principled criterion which presents strong signal to identify irresponsible labelers online. (3) We demonstrate that through explicit modeling of the label consistency in the active learning model, our collaborative active learning process is robust to label noises and label errors from irresponsible labelers. (4) We apply the proposed collaborative active learning framework to learn classifiers for visual recognition, which produced models that can often generalize better to new data than other competing methods.

The remainder of the paper is organized as follows: Sec. 2 presents the mathematical formulation of our collaborative discriminative learning framework. Then in Sec. 3, we develop the active learning criteria for each labeler. In Sec. 4, we derive a principled measure from our computational model to detect irresponsible labelers for label quality control. Various experimental results are reported and discussed in Sec. 5. Finally, we conclude in Sec. 6.

2. Collaborative discriminative learning

2.1. Formulation

Suppose we have K labelers (a.k.a, K Turks in Amazon Mechanical Turk) subscribed to our visual labeling task on data-set $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$. We partition $\mathcal{D}$ into K subsets that have overlaps with each other, i.e., $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2 \cup \ldots \cup \mathcal{D}_K$. Usually we may want to ensure that m versions of the label for each data $\mathbf{x}_i \in \mathcal{D}$ for the target visual concept be collected from different labelers. Hence $\mathbf{x}_i$ will be present in m subsets of $\mathcal{D}$. In other words, define $\mathcal{S}(\mathbf{x}_i) = \{\mathcal{D}_k | \mathbf{x}_i \in \mathcal{D}_k\}$ to be the set of all subset $\mathcal{D}_i$ that $\mathbf{x}_i$ belongs to, we have, $\forall \mathbf{x}_i$, $|\mathcal{S}(\mathbf{x}_i)| = m$, where $|\cdot|$ denotes the cardinality of a set.

Since our goal is to design a collaborative active learning strategy across all the K labelers, we further assume that each subset $\mathcal{D}_i$ is composed of two subsets: the labeled set $\mathcal{L}_i$, and the unlabeled set $\mathcal{U}_i$ such that $\mathcal{D}_i = \mathcal{L}_i \cup \mathcal{U}_i$ and $\mathcal{L}_i \cap \mathcal{U}_i = \emptyset$. We denote $y_i(k) \in \{-1, 1\}$ to be the label of $\mathbf{x}_i$ by labeler k if it is a labeled data sample. Note here, we focus our discussion on binary classification problems but it is straightforward to extend it to multiple category classification by taking a one versus all approach. For each dataset $\mathcal{D}_i$, we try to learn an individual classification function $f_i(\mathbf{x})$, $i = 1, 2, \ldots, K$ from $\mathcal{L}_i$. Notice that the training of the set of all classifiers is not independent, as we would like to ensure that two classifiers $f_i(\mathbf{x})$ and $f_j(\mathbf{x})$ be consistent on the data samples they share.

Therefore, we propose the following objective function

to jointly optimize all K classifiers, i.e.,

$$\begin{aligned} L(\mathcal{D}) &= \sum_{i=1}^{K} \sum_{\mathbf{x}_j \in \mathcal{L}_i} L_i(y_j(i), f_i(\mathbf{x}_j)) \\ &+ \sum_{1 \le i \ne j \le K} \sum_{\mathbf{x}_k \in \mathcal{D}_i \cap \mathcal{L}_j} L_{ij}^l(y_k(j), f_i(\mathbf{x}_k)) \\ &+ \lambda \sum_{i=1}^{K} \Omega(\|f_i\|_{\mathcal{H}}), \end{aligned} \tag{1}$$

where $\Omega(\cdot)$ is a monotonically increasing regularization function to control the complexity of the hypothesis space, and $\mathcal{H}$ is the reproducing kernel Hilbert space induced by certain kernel function. Furthermore, here $L_i(\cdot)$ is a loss function to characterize the performance of each individual classifier $f_i(\mathbf{x}_i)$ on each $\mathcal{L}_i$; L_{ij}^l reinforces that the two classifiers $f_i(\mathbf{x})$ and $f_j(\mathbf{x})$ be consistent in predicting the label of a shared data sample $\mathbf{x}_k$, when it has already been labeled by at least labeler j. For $L_i(\cdot)$, we took a standard Logistic regression loss to maximize the margin, i.e., $L_i(y_j(i), f_i(\mathbf{x}_j)) = \log\{1 + e^{-y_j(i) f_i(\mathbf{x}_j)}\}$.

To define $L_{ij}^l(\cdot)$, we need to consider three conditions. First, if $\mathbf{x}_k \in \mathcal{L}_i \cap \mathcal{L}_j$, i.e., $\mathbf{x}_k$ is labeled by both labeler i and labeler j, and the labels are consistent with each other. Then we would need to bias the learning of both $f_i(\mathbf{x})$ and $f_j(\mathbf{x})$ to make more efforts to ensure the correctness of their prediction on this data sample $\mathbf{x}_k$. If the two labels are inconsistent, then it could either be the case that this example caused confusion among the different labelers, or some labelers are not doing a good job. In this case, we may discount these conflicting labels by encouraging both classifiers $f_i(\mathbf{x})$ and $f_j(\mathbf{x})$ to put the data sample to be near the decision boundary since we are not sure about the true label anyway. In the third case, $\mathbf{x}_k$ is only labeled by labeler j, then this label information will need to be leveraged to benefit the learning of $f_i(\mathbf{x})$. As can be easily verified, we can achieve the desired behavior for all three situations through a single loss function, i.e., $L_{ij}^l(y_k(j), f_i(\mathbf{x}_k)) = \log\{1 + e^{-y_k(j) f_i(\mathbf{x}_k)}\}$.

2.2. Learning a kernel machine

We exploit the "kernel tricks" to learn classifiers with complex decision boundaries, which implicitly performs a non-linear mapping to transform the data in the original space to a very high dimensional space (or even infinite dimensional space). According to the representation theorem [9], each classifier $f_i(\mathbf{x})$, $i = 1, 2, \ldots, K$ is defined as

$$f_i(\mathbf{x}) = \sum_{\mathbf{x}_j \in \mathcal{D}_i} \alpha_{ij} \mathbf{k}(\mathbf{x}_j, \mathbf{x}). \tag{2}$$

Let $N_i = |\mathcal{D}_i|$, $N_i^l = |\mathcal{L}_i|$, and $N_i^u = |\mathcal{U}_i|$ be the number of samples in $\mathcal{D}_i$, $\mathcal{L}_i$ and $\mathcal{U}_i$ respectively. We immediately have $N_i = N_i^l + N_i^u$. We denote $\vec{\alpha}_i = [\alpha_{i1}, \alpha_{i2}, \ldots, \alpha_{iN_i}]^T$. Let $\mathbf{K}_i = [\mathbf{k}(\mathbf{x}_j, \mathbf{x}_k)]_{jk}$ be the $N_i \times N_i$ Gram matrix defined over $\mathcal{D}_i$. Let $\mathbf{K}_i^l = [\mathbf{k}(\mathbf{x}_j, \mathbf{x}_k)]_{\mathbf{x}_j \in \mathcal{L}_i}$ be the first N_i^l rows of $\mathbf{K}_i$, and $\mathbf{K}_i^u = [\mathbf{k}(\mathbf{x}_j, \mathbf{x}_k)]_{\mathbf{x}_j \in \mathcal{U}_i}$ be the last N_i^u rows of $\mathbf{K}_i$, i.e., $\mathbf{K}_i = [\mathbf{K}_i^{lT}, \mathbf{K}_i^{uT}]^T$. We further denote that $\mathbf{K}_{ij}^l$ be the matrix composed by rows of $\mathbf{K}_i$ corresponding to those samples $\mathbf{x}_k \in \mathcal{D}_i \cap \mathcal{L}_j$. Similarly we denote $\mathbf{K}_{ij}^u$ and $\mathbf{K}_{ji}^u$ be the matrices composed by rows of $\mathbf{K}_i$ and $\mathbf{K}_j$ corresponding to those data samples in $\mathcal{U}_i \cap \mathcal{U}_j$, respectively.

We also denote that $\forall i$, $\mathbf{y}_i$ be the label vectors of the set of labeled data samples in $\mathcal{L}_i$ from labeler i, and $\mathbf{y}_{ij}^l$ be the label vector of those samples in $\mathcal{D}_i \cap \mathcal{L}_j$ from labeler j. Embedding Eq. 2 into Eq. 1, and representing the formula in vector format, we have

$$\begin{aligned} L(\mathcal{D}) = &\sum_{i=1}^{K} \mathbf{1}^T \log\{1 + e^{-\mathbf{K}_i^l(\mathbf{y}_i)\vec{\alpha}_i}\} \\ &+ \textstyle\sum_{i \ne j} \mathbf{1}^T \log\{1 + e^{-\mathbf{K}_{ij}^l(\mathbf{y}_{ij}^l)\vec{\alpha}_i}\} + \lambda \sum_i \vec{\alpha}_i^T \mathbf{K}_i \vec{\alpha}_i, \end{aligned} \tag{3}$$

where $\mathbf{K}_i^l(\mathbf{y}_i) = diag[\mathbf{y}_i]\mathbf{K}_i^l$ and $\mathbf{K}_{ij}^l(\mathbf{y}_{ij}^l) = diag[\mathbf{y}_{ij}^l]\mathbf{K}_{ij}^l$. Here $diag[\mathbf{v}]$ transforms a vector into a diagonal matrix by placing each corresponding element of the vector $\mathbf{v}$ sequentially in the diagonal position to form a diagonal matrix.

It can be shown that $L(\mathcal{D})$ is a convex function with respect to each $\vec{\alpha}_i$. Hence we can conveniently obtain the optimal solution of $\vec{\alpha}_i$ by gradient based optimization algorithms. We have

$$\frac{\partial L(\mathcal{D})}{\partial \vec{\alpha}_i} = -\mathbf{K}_i^l(\mathbf{y}_i)^T \mathbf{P}_i^l - \sum_{i \ne j} \mathbf{K}_{ij}^l(\mathbf{y}_{ij}^l) \mathbf{P}_{ij}^l + 2\lambda \mathbf{K}_i \vec{\alpha}_i, \tag{4}$$

where $\mathbf{P}_i^l = \frac{e^{-\mathbf{K}_i^l(\mathbf{y}_i)\vec{\alpha}_i}}{1+e^{-\mathbf{K}_i^l(\mathbf{y}_i)\vec{\alpha}_i}}$ and $\mathbf{P}_{ij}^l = \frac{e^{-\mathbf{K}_{ij}^l(\mathbf{y}_{ij}^l)\vec{\alpha}_i}}{1+e^{-\mathbf{K}_{ij}^l(\mathbf{y}_{ij}^l)\vec{\alpha}_i}}$. Although we can take the second order derivative to run a full Newton's method, we resort to a more efficient quasi-Newton's method, i.e., the L-BFGS-B algorithm [27] to obtain the optimal $\vec{\alpha}_i$ more efficiently.

Discussions. It can be clearly observed that the learning of classifier f_i will take into consideration of labels from other labelers with shared data. This is by design from our collaboration formulation, as labels are naturally shared across different labelers on the shared data. As we will show in the experiments, this gives an additional advantage to enable the learning to progress faster than independently learning multiple classifiers.

2.3. Kernel machine ensemble

Once all the kernel classifiers $f_i(\mathbf{x})$ are learnt, to classify a new data point $\mathbf{x}_{new}$, we took an ensemble classification approach. Specifically, we identify the nearest neighbor $\mathcal{N}(\mathbf{x}_{new})$ of $\mathbf{x}_{new}$ in $\mathcal{D}$. The final prediction of $\mathbf{x}_{new}$ is determined by the following ensemble classifier

$$f(\mathbf{x}_{new}) = \sum_{\mathcal{N}(\mathbf{x}_{new}) \in \mathcal{D}_i} f_i(\mathbf{x}_{new}), \tag{5}$$

where $\mathbf{D}_i$ indicates the subset of the training data assigned to labeler i to learn $f_i(\mathbf{x})$. Since each data sample is assigned to m labelers, so there will be exactly m learned

kernel classifiers to be used to form the ensemble classifier to predict any new data sample $\mathbf{x}_{new}$. Alternatively, we can also sum the prediction scores from all K classifiers together. Empirically we found the ensemble classifier in Eq. 5 always obtained better results.

3. Collaborative active learning

We design a collaborative active learning strategy based on the collaborative discriminative kernel machine proposed in Sec. 2.2. Recall that for each single labeler i, the task of active learning is to select the most informative example $\mathbf{x}_k \in \mathcal{U}_i$ to be labeled by the labeler, such that the performance of the learning machine can be improved the most. One natural criterion is to evaluate how far the unlabeled example $\mathbf{x}_k \in \mathcal{U}_i$ is from the decision boundary with the currently learnt classifier

$$f_i(\mathbf{x}_k) = \sum_{\mathbf{x}_j \in \mathcal{D}_i} \alpha_{ij} \mathbf{k}(\mathbf{x}_j, \mathbf{x}_k). \tag{6}$$

If the absolute value of $f_i(\mathbf{x}_k)$ is small, then it indicates that our current classifier is not very confident with it. Hence, it is natural for us to define our active learning criterion for labeler i to be

$$\mathcal{A}_i(\mathbf{x}_k) = |f_i(\mathbf{x}_k)|. \tag{7}$$

At each round of the active learning step, we choose

$$\mathbf{x}_i^* = \arg\min_{\mathbf{x}_k \in \mathcal{U}_i} \mathcal{A}_i(\mathbf{x}_k) \tag{8}$$

for labeler i to label. This type of uncertainty based active sampling, though simple, has been demonstrated to be very effective in previous work. Certainly, more complicated active learning criterion can be adopted at the expense of more computational cost. It is beyond the scope of this paper to explore all of them. In our experiments, it is revealed that this simple uncertainty based active learning criterion performed very well.

We also would like to emphasize that although our active learning criterion $\mathcal{A}_i(\mathbf{x})$ for labeler i is derived from the classification function $f_i(\mathbf{x})$ only, it does not mean that the active example selection is independent of each labeler. That is because the learning of each $f_i(\mathbf{x})$ is coupled with each other in our joint formulation (Eq. 1 and Eq. 3). Therefore the dependent information from other labelers have been carried over into the active selection criterion. Moreover, as clearly presented in our formulation, once $\mathbf{x}_i^*$ is selected, it will also affect the learning of the classifiers of the other labelers. Hence the active sample selection processes of all the K labelers are indeed coupled with one another in our formulation.

Each time a new image or several new images are labeled by the labelers, the $f_i(\mathbf{x})$ for each specific labeler i needs to be updated. We shall note that in our collaborative learning framework, the update or retraining of $f_i(\mathbf{x})$, or equivalently the re-estimation of the parameter vector $\vec{\alpha}_i$ can run asynchronously – we simply just need to hold the classifier parameters $\vec{\alpha}_j$ of the other labelers to be fixed when calculating the gradient using Eq. 4, and then optimizing for $\vec{\alpha}_i$ only.

4. Labeling quality control

Most previous collaborative tagging systems such as Amazon Mechanical Turk can only rely on post check of label consistency to filter out noisy labels. By that time, even if a sloppy labeler was identified, valuable time and monetary resources have been wasted. We argue that the consistency among the learned kernel machines $f_i(\mathbf{x}_i)$ can naturally serve as an online label quality indicator. As we have discussed in Sec. 2.1, when the labels from two labelers i and j on an example $\mathbf{x}_k$ are conflicting with each other, our joint formulation will encourage the classifier $f_i(\mathbf{x}_k)$ and $f_j(\mathbf{x}_k)$ to have low confidence predictions on $\mathbf{x}_k$. Hence we define the following evaluation function to indicate if labeler i is consistently conflicting with other labelers, i.e.,

$$Q_i = \frac{1}{|\mathcal{L}_i|} \sum_{\mathbf{x}_j \in \mathcal{L}_i} y_j(i) f_i(\mathbf{x}_j). \tag{9}$$

Intuitively, if labeler i is doing a lousy job in labeling, then it will induce more conflicts with its peers and its Q_i score will be low. Although the Q score of the other labelers will also be degraded by labeler i's irresponsible behavior, they will be degraded less than the Q score of labeler i. Nevertheless, we are still assuming that the majority of the labelers will behave honestly–as is the case in real-world.

5. Experiments

5.1. Datasets and visual features

We start our evaluation with a set of experiments with controlled synthetic label noises on 10 different classes of images from the ImageNet dataset to better understand its behavior. Then we evaluate it on two datasets with real-world crowdsourced labels from Amazon Mechanical Turk and compare with some previous work.

For the experiments with synthetic label noise, we leverage images in 10 different classes from the ImageNet dataset [5]. These are top 10 classes with the largest number of labeled examples from ImageNet Challenge. The category names of the 10 classes of images are "seashore, coast, seacoast, sea-coast", "monarch, monarch butterfly, milkweed butterfly, Danaus plexippus", "Vizsla, Hungarian pointer", "English setter", "Yorkshire terrier", "Rhodesian ridgeback", "African elephant, Loxodonta africana", "meerkat, mierkat", "computer keyboard, keypad", "dining table, board" , respectively. The number of images per category for these 10 categories used for collaborative active learning ranges from 2125 to 3047. There are 24084 images in total. Note these accounted for 80% of the labeled images for these 10 categories in ImageNet dataset. We hold the other 20% for testing the resulting ensemble classifiers. In terms of visual features, we used the local coordinate coding (LCC) [15] on dense HoG features with 4096 codewords, and spatially pooled the LCC features in 10 spatial cells. This is similar to [15]. The dimensionality of the features is 40960.

For the experiments with **real** crowdsourced labels, we re-pushed the images of the 5 categories "Yorkshire terrier",

"Rodesian ridgeback", "English setter", "Vizsla Hungarian pointer", and "Meerkat, meerkat" back to Amazon Mechanical Turk to collect multiple copies of labels. The first 4 categories are all different breed of dogs, and the last category "Meerkat, meerkat" is similar in visual appearance to dogs. Therefore these 5 categories tend to confuse with one another. We obtain 7 copies of labels per image for each image in these 5 categories, which are subsequently used in our experiments. The noise level of the labels we obtained for each category varies. The percentage of the labels being correct for these five visual categories are 94.96%, 68.91%, 87.01%, 68.43%, and 98.01%, respectively.

Another datasets with **real** crowdsourced labels we experimented is a face dataset for a gender recognition problem. Through Amazon Mechanical Turk, we have collected 5 copies of labels (male/female) for 9441 face images. We hold out 2000 of face images which had all 5 copies of labels in consensus for testing purpose and the rest of the face images with different percentage of label inconsistency are used for collaborative active learning. The face images are all 64×64, from each of which we extract a 5408 dimensional discriminative features. This feature is the output from the last layer of a convolutional neural network trained for gender recognition with a separate small set of labeled gender face images. We will make both the features and labels of these two datasets publicly available upon publication of this paper.

Performance evaluation. We conduct experiments on these datasets to measure how our proposed method and other competing methods are performing with the progress of the active learning process. Note that at each learning step, the different competing methods may add different number of labels, instead of plotting the progression of the recognition accuracy w.r.t. the active learning steps, we plot the progression of the recognition accuracy w.r.t. the number of labels to ensure a fair comparison in all figures.

5.2. Experiments with synthetic label noise

Efficacy of collaborative active learning. For evaluation, for each of the 10 image category from ImageNet Challenge, we randomly sample an equal number of images from the other 9 categories to serve as its negative images. We ensure that each image will be assigned to $m = 5$ labelers. We distributed the training data evenly to 20 labelers to ensure that roughly 1000 images are allocated to each labeler.

We run simulation experiments with the proposed collaborative active learning algorithm and compare it with five baseline algorithms. The first baseline algorithm uses the same discriminative formulation in Eq. 1 and Eq. 3 but only randomly selects the next image to be labeled for each labeler. The second baseline algorithm is to run multiple independent active learning process with the proposed kernel machine in Sec. 2.2. It is equivalent to discarding the cross labeler loss function $L^l_{ij}(\cdot)$ in Eq. 1, which corresponds to the middle term in Eq. 3. The active learning criterion for it is in the same form as Eq. 8. The third baseline algorithm is training multiple independent discriminative classifiers in the same way as the second baseline algorithm, but selecting the images to be labeled next in a random fashion.

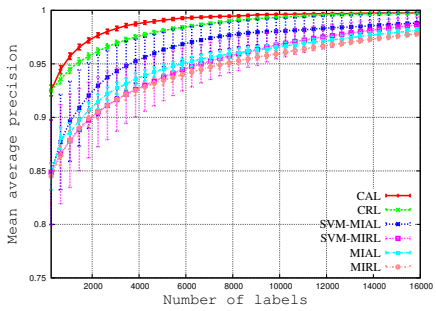
In addition, we also run multiple independent active learning SVM and multiple independent random learning SVM, respectively, which is similar to the previous two baseline algorithms using hedge loss instead of logistic regression loss. For notation simplification, we denote our proposed collaborative active learning algorithm to be CAL. We further denote the first to five baseline algorithms to be CRL, MIAL, MIRL, SVM-MIAL, and SVM-MIRL, respectively.

We present the experimental results on Fig. 1. The results on the active learning pool and the hold-out test dataset are presented in Fig. 1a, and 1b, respectively. In both figures, the horizontal axis shows the number of labels added in the labeling process. In Fig. 1a, the vertical axis represents the mean average precision (mAP) (the mean is taken over all the runs of 10 categories from all labelers) of the learned classifiers over the examples in the active learning pool. In Fig. 1b, the vertical axis represents the mAP of the learned ensemble classifiers on the hold-out testing datasets, which are also averaged over all the 10 categories. Therefore, at each step, each labeler is providing label for one data sample.

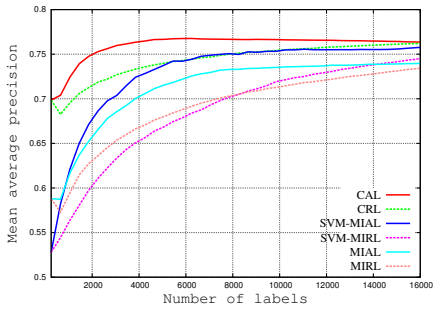
We adopt average precision (AP) as the criterion to give a more comprehensive evaluation of the classifiers. So the different curves reflect how the mAP evolves with our method and the other five baseline algorithms, respectively. All figures clearly show that exploiting active learning to select samples is often better than selecting the samples randomly. This is exemplified by the fact that the recognition curve of CAL is always higher than CRL, and the recognition curve of MIAL is always higher than MIRL. With the active sample selection, we can achieve higher mAP in recognition sooner with fewer labeled images than using random sample selection. We only show the average results across all labelers over all image categories due to the space limit. The figures on each individual category consistently presented the same trend. We omit them due to space limit.

In particular, the mAP curve of CAL is always higher than MIAL, which validated the efficacy of our collaborative formulation. By ensuring the consistencies among the classification models through the shared data, our collaborative discriminative learning paradigm allows the label information to be shared among labelers and hence better utilize them to train better classifiers. Since the only difference between CAL and MIAL is the cross labeler cost terms defined in Eq. 1 and Eq. 2, it is clear that it is the collaborative formulation really leads to the improvement.

Specifically, in our collaborative formulation, if labeler A labeled a data sample shared by labeler B. That label is immediately factored into the learning of the classifier of labeler B even if B has not labeled it yet. For MIAL, there is no cross labeler cost so the label is not shared. We also want to point out that the ensemble classifier produced by CAL always achieved better accuracy in the held-out testing dataset, which implies that our collaborative formulation can help learn classifiers that can generalize well. Note in all our experiments, we start evaluating the recognition accuracy from 50 labeled images.

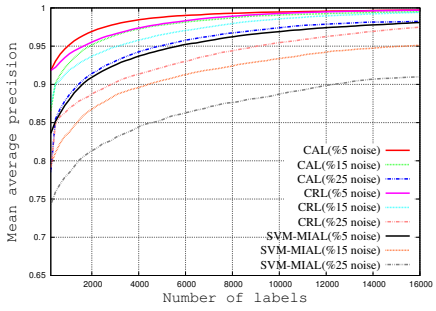


(a) Active learning pool

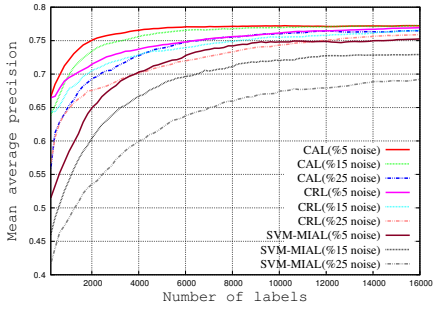


(b) Hold-out testing dataset

Figure 1: Recognition performance with clean labels without noise. The vertical bar indicates the standard deviation of mAP values on the curve.

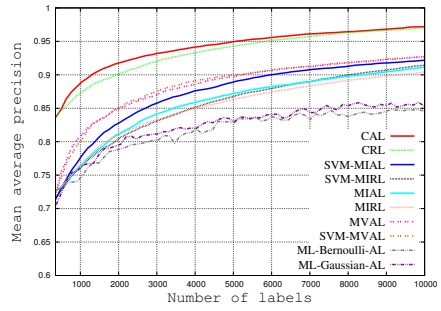


(a) Active learning pool

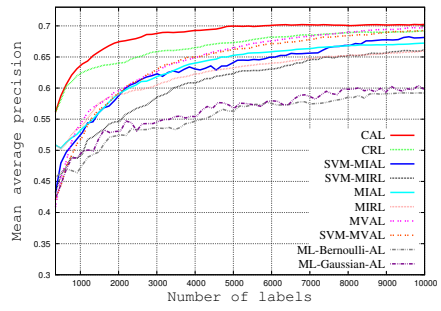


(b) Hold-out testing dataset

Figure 2: Recognition performance on the active learning pool with different levels of label noises, and the hold-out testing dataset, respectively.



(a) Active learning pool

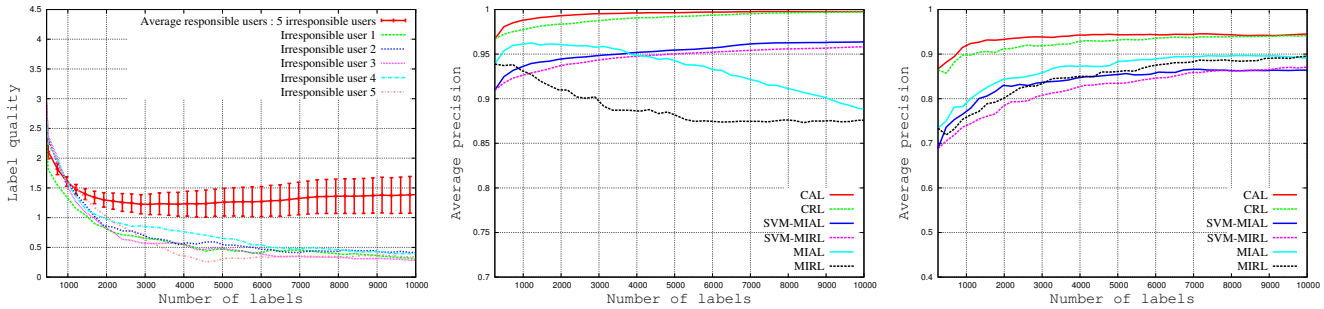


(b) Hold-out testing dataset

Figure 3: Recognition performance with real crowd-sourced labels on five ImageNet categories in active learning pool and hold-out testing dataset.

Different noise levels. To demonstrate that our proposed CAL algorithm is indeed more robust to label noise. We simulate the case that the labelers have a chance to generate noisy labels, ranging from 5%, 15%, to 25%, meaning that the labeler has such a probability to label the image incorrectly. We run the experiments with different level of label noises for all 20 labelers on all the 10 image classes. Three methods are compared, i.e., our proposed CAL, the CRL, and the SVM-MIAL (MIAL is always inferior to SVM-MIAL). As we can observe from Fig. 2, the general trend is that the performances of the classifiers all drop with the increase of label noise levels. However, at all noise levels, our proposed CAL algorithm always achieves better mAP scores on both the active learning pool (Fig. 2a) and the hold-out testing dataset (Fig. 2b) across the learning process. Hence it provides solid evidence that our proposed collaborative learning framework can largely suppress the negative effects of the noisy labels. The curve is averaged over the 10 categories over all labelers. The curves under 35% label noises showed similar phenomenon, we omitted it for a more clean view.

Detection of irresponsible labelers. In this section, we intend to demonstrate that our label quality measure (Eq. 9) can readily capture irresponsible labelers. We show the experimental results in Fig. 4. The experiments are performed on the "Meerkat, meerkat" class of the ImageNet dataset with 22 labelers. In Fig. 4a, we simulate the case that there are 5 irresponsible labelers and the rest are responsible labelers (with 5% label noise). It is clearly observed that the label quality of the irresponsible labelers are consistently and significantly lower than those of the responsible labelers across the collaborative active learning process. It falls below the lower variance bar of the label quality of the responsible labelers. This is a very strong and consistent signal that enables us to capture irresponsible labelers from the very beginning of the collaborative active labeling process. This validated our hypothesis that the average of signed classification scores on the labeled images for each labeler (as defined in Eq. 9) naturally serves as a label quality measure to detect irresponsible labelers. In Fig. 4b and 4c, we compare the AP scores of our CAL algorithm with those of SVM-MIAL, SVM-MIRL, MIAL, and MIRL under the presence of five irresponsible labelers on the active learning pool and hold-out testing dataset, respectively. It is clear that our proposed CAL algorithm is more robust to the presence of irresponsible labelers. The AP of MIAL and MIRL on the active learning pool actually dropped when more labels are added due to the bad performance of the classifiers from those 5 irresponsible labelers. However, the SVM-MIAL and SVM-MIRL do not suffer from this in the active learning pool, suggesting that the hedge loss is more robust. We will consider the adoption of hedge loss in a collaborative formulation in our future work. We have also run extensive experiments on all 10 image categories with different number irresponsible labelers (upto half), and the observations are consistent with what we show in Fig. 4.



(a) Label quality with 5% label noise for responsible labelers and 5 irresponsible labelers.

(b) AP with 5% label noise and five irresponsible labelers on active learning pool.

(c) AP with 5% label noise and five irresponsible labelers on hold-out testing dataset.

Figure 4: The label quality and recognition accuracy of responsible labelers are averaged with variance bar overlayed. The irresponsible labelers are plotted alone.

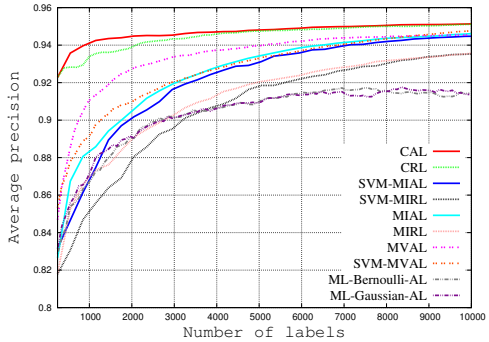
5.3. Experiments with real crowd-sourced labels

In this section, we conduct experiments on two datasets with real crowdsourced labels from Amazon Mechanical Turk. In addition to comparing with the original 5 baseline algorithms, we add two new baseline algorithms and also compare with two version of the active learning algorithms presented in Yan *et al.* [24, 25]. The two new baseline algorithms adopt an online majority voting strategy in the active learning process to induce a single kernel classifier using either the logistic regression loss (as in our formulation) or hedge loss (as in SVM). Specifically, at each round of the active learning step, each data sample is labeled by 7 or 5 labelers, and we utilize the majority voted label as the label for this data sample and re-train the classifier. We name these two baseline algorithms as MVAL and SVM-MVAL, respectively. The two algorithms presented in Yan *et al.* [24, 25] are named as ML-Bernoulli-AL and ML-Gaussian-AL, respectively, according to two differen probability distribution they exploited in their model.

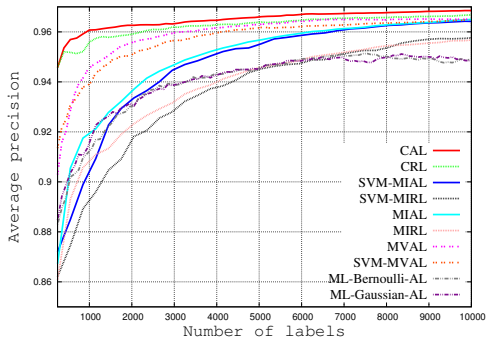
We want to clarify that for MVAL and SVM-MVAL, the active learning pool contains all images, so it is a larger pool than the pool of examples handled by each individual labeler in our CAL formulation. However, our comparison is still fair because the horizontal axis in the figure indicates the total number of labels added in the learning process.

Five categories of ImageNet. We followed the same data split for active learning and hold-out testing as in the experiments with synthetic noisy label. Each image is assigned to $m = 7$ labelers as we have seven copies of crowdsourced labels per image. Hence 14 to 21 classifiers per category trained with the real crowdsourced label depending on the number of images each category has, to compose the final ensemble classifier. Fig. 3 presents the mAP curves on the active learning pool and the hold-out testing dataset. Our proposed CAL outperformed all the other 9 competing algorithms in both the active learning pool and the hold-out testing datasets.

In particular, we want to highlight the comparison with the four new competing methods, as they all induce a single strong classifier for recognizing each category. The fact that they show inferior recognition accuracy compared with CAL is a strong indication that ensemble classifier learning is more robust to label noises. The two methods proposed by Yan *et al.* [24, 25] compete even less favorably than the MVAL and SVM-MVAL baselines. A potential reason is that their methods tend to make over-confident predictions.



(a) Active learning pool



(b) Hold-out testing dataset

Figure 5: Recognition performance on real crowd-sourced labels on a face gender image dataset. The vertical bar indicates the standard deviation of AP values on the curve.

Gender face dataset. Fig. 5 presents the experimental results of running our CAL algorithm and all other 9 competing methods on the gender face image dataset. Each data sample is assigned to $m = 5$ labelers to label and around thirty labelers in total are used. Again, it is clear that our proposed CAL algorithm outperformed all other 9 competing methods in both the active learning pool and the hold-out testing datasets. The results also demonstrated the efficacy of our collaborative model formulation, as the second

best algorithm is CRL while the other algorithms are either running multiple independent processes for model learning or just inducing a single classifier using active learning. Again, the AP on the active learning pool is the mean across all labelers, while the AP on the hold-out testing dataset is computed using the resulting ensemble kernel classifier. The observation is consistent with our experiments on ImageNet dataset with real crowdsourced labels.

6. Conclusion and future work

In view of the popularity of using crowd-sourcing tools for labeling large scale image datasets for research on visual recognition, and to mitigate issues in existing crowdsourcing tools, we present a collaborative active learning framework to support multiple labelers to collaboratively label a set of images to learn an ensemble kernel machine classifier. As verified by our experiments, our approach enables more efficient model learning from multiple labelers, is robust to label noise and irresponsible labelers, and can readily detect irresponsible labelers online. Our future work includes extending the proposed framework to handle multiple target labeling tasks. We also plan to implement it in a cloud computing environment. Once these are fulfilled, we will deliver an end-to-end service to support any large scale multi-labeler interactive model learning efforts.

Acknowledgement

This work is partly supported by US National Science Foundation Grant IIS 1350763, China National Natural Science Foundation Grant 61228303, GH's start-up funds from Stevens Institute of Technology, a Google Research Faculty Award, a gift grant from Microsoft Research, and a gift grant from NEC Labs American.

References

[1] V. Ambati, S. Vogel, and J. Carbonell. Active learning and crowd-sourcing for machine translation. In N. C. C. Chair), K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may 2010. European Language Resources Association (ELRA). 1

[2] S. Chen, J. Zhang, G. Chen, and C. Zhang. What if the irresponsible teachers are dominating? a method of training on samples and clustering on teachers. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*. American Association of Artificial Intelligence, July 2010. 2

[3] O. Dekel and O. Shamir. Good learners for evil teachers. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 233–240, New York, NY, USA, 2009. ACM. 2

[4] O. Dekel and O. Shamir. Vox populi: Collecting high-quality labels from a crowd. In *In Proceedings of the 22nd Annual Conference on Learning Theory*, 2009. 2

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 248 –255, June 2009. 1, 4

[6] P. Donmez, J. Carbonell, and J. Schneider. A probabilistic framework to learn from multiple annotators with time-varying accuracy. In *Proceedings of the SIAM International Conference on Data Mining (SDM 2010)*, pages 826–837, 2010. 2

[7] P. Donmez, J. G. Carbonell, and J. Schneider. Efficiently learning the accuracy of labeling sources for selective sampling. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 259–268, New York, NY, USA, 2009. ACM. 1

[8] S. Ebert, M. Fritz, and B. Schiele. Ralf: A reinforced active learning formulation for object class recognition. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3626 –3633, june 2012. 1

[9] I. Guyon, B. Boser, and V. Vapnik. Automatic capacity tuning of very large vc-dimension classifiers. In *Advances in Neural Information Processing Systems*, 1993. 3

[10] S. C. Hoi, W. Liu, and S.-F. Chang. Semi-supervised distance metric learning for collaborative image retrieval. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1 –7, June 2008. 1

[11] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell. Active learning with gaussian processes for object categorization. In *IEEE International Conf. on Computer Vision*, pages 1–8, October 2007. 1

[12] A. Kovashka, S. Vijayanarasimhan, and K. Grauman. Actively selecting annotations among objects and attributes. In *IEEE International Conference on Computer Vision*, pages 1403–1410, November 2011. 1

[13] A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems 26 (NIPS)*, Lake Tahoe, CA, December 2012. 1

[14] F. Laws, C. Scheible, and H. Schütze. Active learning with amazon mechanical turk. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1546–1556, 2011. 1

[15] Y. Lin, F. Lv, S. Zhu, M. Yang, T. Cour, K. Yu, L. Cao, and T. Huang. Large-scale image classification: Fast feature extraction and svm training. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1689 –1696, June 2011. 4

[16] C. Loy, T. Hospedales, T. Xiang, and S. Gong. Stream-based joint exploration-exploitation active learning. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1560 – 1567, june 2012. 1

[17] V. C. Raykar and S. Yu. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *J. Mach. Learn. Res.*, 13:491–518, Mar. 2012. 2

[18] V. C. Raykar, S. Yu, L. H. Zhao, A. Jerebko, C. Florin, G. H. Valadez, L. Bogoni, and L. Moy. Supervised learning from multiple experts: whom to trust when everyone lies a bit. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 889–896, New York, NY, USA, 2009. ACM. 2

[19] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image. *International Journal of Computer Vision*, 77(1-3):157–173, May 2008. 1

[20] J. Sanchez and F. Perronnin. High-dimensional signature compression for large-scale image classification. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1665 –1672, june 2011. 1

[21] A. Vezhnevets, J. Buhmann, and V. Ferrari. Active learning for semantic segmentation with expected change. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3162 –3169, june 2012. 1

[22] S. Vijayanarasimhan and K. Grauman. Large-scale live active learning: Training object detectors with crawled data and crowds. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1449–1456, june 2011. 1

[23] P. Welinder and P. Perona. Online crowdsourcing: Rating annotators and obtaining cost-effective labels. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 25 –32, june 2010. 1

[24] Y. Yan, R. Rosales, G. Fung, and J. Dy. Active learning from crowds. In *Proc. International Conference on Machine Learning*, 2011. 2, 7

[25] Y. Yan, R. Rosales, G. Fung, and J. Dy. Active learning from multiple knowledge sources. In *Proc. International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2012. 2, 7

[26] L. Zhao, G. Sukthankar, and R. Sukthankar. Incremental relabeling for active learning with noisy crowdsourced annotations. In *Privacy, security, risk and trust (passat), 2011 ieee third international conference on and 2011 ieee third international conference on social computing (socialcom)*, pages 728 –733, oct. 2011. 2

[27] C. Zhu, R. H. Byrd, P. Lu, and J. Nocedal. Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization. *ACM Transactions on Mathematical Software*, 23:550–560, December 1997. 3