Active Learning with Imbalanced Multiple Noisy Labeling

Jing Zhang, *Student Member, IEEE*, Xindong Wu, *Fellow, IEEE*, and Victor S. Sheng, *Member, IEEE*

***Abstract*—With crowdsourcing systems, it is easy to collect multiple noisy labels for the same object for supervised learning. This dynamic annotation procedure fits the active learning perspective and accompanies the imbalanced multiple noisy labeling problem. This paper proposes a novel active learning framework with multiple imperfect annotators involved in crowdsourcing systems. The framework contains two core procedures: label integration and instance selection. In the label integration procedure, a positive label threshold (PLAT) algorithm is introduced to induce the class membership from the multiple noisy label set of each instance in a training set. PLAT solves the imbalanced labeling problem by dynamically adjusting the threshold for determining the class membership of an example. Furthermore, three novel instance selection strategies are proposed to adapt PLAT for improving the learning performance. These strategies are respectively based on the uncertainty derived from the multiple labels, the uncertainty derived from the learned model, and the combination method (CFI). Experimental results on 12 datasets with different underlying class distributions demonstrate that the three novel instance selection strategies significantly improve the learning performance, and CFI has the best performance when labeling behaviors exhibit different levels of imbalance in crowdsourcing systems. We also apply our methods to a real-world scenario, obtaining noisy labels from Amazon Mechanical Turk, and show that our proposed strategies achieve very high performance.**

***Index Terms*—Active learning, crowdsourcing, imbalanced learning, repeated labeling, supervised classification.**

I. Introduction

In many real-world applications, the labels of training examples are acquired dynamically with high cost. It is better to acquire labels for fewer but more valuable examples to achieve higher learning performance. Active learning [19] was proposed to address this issue. The key idea behind active learning is that a learning algorithm is allowed to actively acquire data to improve its learning goal. For example, an active learner aims to achieve high classification accuracy by selecting a subset of most critical instances, and acquiring their labels, thereby reducing the total cost of acquiring labels [20]. In most traditional active learning scenarios, there exists an expert labeler (oracle) to provide the ground truths for unlabeled examples when the learner queries the labels for these examples. In crowdsourcing systems, multiple weak nonexpert labelers annotate the same example, and the integrated label of an example is induced according to the consensus of these different nonexpert labelers.

The simplest and most efficient label integration method is majority voting (MV). It works well under the circumstance that the information about labeling qualities of labelers, underlying class distributions and the difficulties of instances is totally unknown, which is called agnostic. Agnostic methods are more attractive in crowdsourcing systems, since crowdsourcing tasks always provide the challenge that prior knowledge is inadequate, especially when the tasks have never been conducted before or the labelers have a strong desire for their own privacy protection. MV works very well under the two prerequisites that: 1) the overall labeling quality of most labelers is larger than 50% in binary labeling tasks and 2) the errors of each labeler are approximately uniformly distributed over all classes. However, these prerequisites do not always hold in complicated real-world applications. Due to lack of expert knowledge, most labelers tend to make shallow determinations by common sense or simply repeat what others say. These difficulties cause imbalanced multiple noisy labeling [31]. Taking binary labeling for example, it is not unusual for labeling on minority examples to be error-prone.

We treat the minority as the positive class in this paper. When labeling behaviors of labelers are imbalanced, the number of negative labels obtained is far more than that of positive ones. If MV is simply applied under this situation, it does not work at all, and would result in an extremely imbalanced class distribution in its final induced training set. This phenomenon is observed in many real-world crowdsourcing datasets collected from Amazon Mechanical Turk. Table I shows four public crowdsourcing datasets [25] that were obviously labeled unevenly. These datasets exhibit the common characteristic that the labeling quality of negative examples is significantly greater than that of positive ones. Although multiple noisy labels are acquired for each example, MV can

Manuscript received December 30, 2013; revised May 20, 2014 and July 16, 2014; accepted July 25, 2014. Date of publication August 14, 2014; date of current version April 13, 2015. This work was supported in part by the Program for Changjiang Scholars and Innovative Research Team in University (PCSIRT) of the Ministry of Education, China, under Grant IRT13059, in part by the National 973 Program of China under Grant 2013CB329604, in part by the National 863 Program of China under Grant 2012AA011005, in part by the National Natural Science Foundation of China under Grant 61229301, and in part by the U.S. National Science Foundation (IIS-1115417). This paper was recommended by Associate Editor Y. S. Ong. *(Corresponding Author: X. Wu.)*

J. Zhang is with the School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230009, China (e-mail: jingzhang.cs@gmail.com).

X. Wu is with the School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230009, China, and also with the Department of Computer Science, University of Vermont, Burlington, VT 05405 USA (e-mail: xwu@cs.uvm.edu).

V. S. Sheng is with the Department of Computer Science, University of Central Arkansas, Conway, AR 72035 USA (e-mail: ssheng@uca.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2014.2344674

TABLE I
STATISTIC INFORMATION OF FOUR DATASETS WITH IMBALANCED LABELING

Dataset	$\overline{p}$	$\overline{p}_N$	$\overline{p}_P$	T_P	L_P	$\overline{\#l}$	acc_{MV}
AC2	74.6	71.5	46.9	43.8	36.5	9.96	84.4
Trec10	67.2	71.1	37.4	45.7	32.1	5.66	64.2
SpamCF	69.4	91.9	5.10	31.0	7.75	23.0	66.0
Valence	65.0	87.3	41.5	50.0	26.4	10.0	68.0

$\overline{p}$, $\overline{p}_N$ and $\overline{p}_P$: average labeling qualities overall, and labeling quality on each of two classes, respectively. T_P and L_P: the percentages of *true* (+) examples and (+) labels obtained. $\overline{\#l}$: average number of labels per example. acc_{MV}: integrated accuracy using MV.

hardly improve their integrated accuracy. That is, MV loses effectiveness under this circumstance. According to binomial theory [12], their integrated accuracies can be ideally improved to 96.3% (*AC2*), 83.5% (*Trec10*), 97.5% (*SpamCF*), and 85.1% (*Valence*). All datasets shown in Table I was originally created for the purpose of studying the integrated quality of multiple noisy labels. Unfortunately, they are not suitable for training learning models.[1] We cannot use them to conduct our experiments in later sections of this paper.

Besides labeling imbalance, which is caused by biases of users' labeling behaviors, the class distribution of a dataset may also be imbalanced. That is, the numbers of examples belonging to two classes are significantly different, which is known as an imbalanced dataset. For example, the proportion of positive examples in the *SpamCF* dataset is only 31% (T_P). On the dataset with an imbalanced class distribution, labeling imbalance will seriously deteriorate the performance of label integration (e.g., MV). For example, acc_{MV} on *SpamCF* is even less than the average quality of individuals ($\overline{p}$). In [32], an extreme case is provided to show that when imbalanced labeling happens on some imbalanced datasets, the minority class (positive) examples diminish after label integration with MV, because each example obtains too many negative noisy labels.

To address the imbalanced multiple noisy labeling in binary supervised learning, [32] proposed an agnostic heuristic algorithm named positive label frequency threshold (PLAT) to deal with the imbalanced labeling issue. PLAT does not require any prior knowledge, and induces the integrated label of an example in the training set only based on its observed multiple noisy labels. It not only effectively deals with the imbalanced multiple noisy labeling, which off-the-shelf agnostic methods cannot cope with, but also performs the same as MV under the circumstance that the labeling behaviors are balanced.

In [32], every instance has a repeated label set with a fixed number of labels. If the training set has a great number of examples, the total number of labels acquired is huge. On one hand, this repertory has high label acquisition cost as every example has the same number of labels regardless of whether this example is important for model learning or not. On the other hand, in many real-world applications, the process of acquiring labels is dynamic, and not all instances can have their potential labels at the same time. Some instances may require more concentration and effort. Thus, it is interesting and necessary to investigate the process of noisy label acquisition in the active learning framework.

This paper focuses on studying active learning with imbalanced multiple noisy labeling. The contributions of this paper are as follows.

1) A novel active learning-based imbalanced multiple noisy labeling framework is proposed for crowdsourcing. In the proposed framework, we combine the label integration and instance selection procedures into a single method, which has not been investigated in previous studies [12], [24].
2) Three novel instance selection strategies, which consider both uncertainty measures and imbalance information, are proposed to improve the performance of active learning. Experimental results show that the proposed three instance selection strategies effectively improve the active learning performance.
3) This paper considers the imbalanced datasets and imbalanced labeling simultaneously. To the best of our knowledge, this is the first effort toward considering the two kinds of imbalances together in crowdsourcing under the active learning paradigms.

II. RELATED WORK

A large number of traditional active learning methods have been proposed to query the membership (class label) of an unlabeled instance directly. These methods can be categorized into three classes based on the problem scenarios: pool-based, stream-based, and query construction-based active learning. Pool-based active learning [20], [26] assumes that there is a small set of labeled data $\mathcal{L}$ and a large pool of unlabeled data $\mathcal{U}$. Queries are selectively drawn from this unlabeled pool which is usually assumed to be closed (i.e., no new unlabeled data will be added into this pool in the future). The learner measures the utility of each example in the pool and selects those examples that can maximally improve the performance of a current model. Stream-based active learning [3], [34] assumes that each unlabeled instance is typically drawn one at a time from the data source in a stream fashion, and the learner must decide whether to query or discard it. The main difference between pool-based and stream-based scenarios is that the former evaluates the entire collection before selecting the best query, whereas the latter makes query decisions individually. Query construction-based active learning [13] generates some synthetic instances, and then queries labels for these instances to extend the labeled training set. It enlarges unlabeled instances to the entire input space rather than those sampled from some underlying natural distribution. Our approach is different from traditional active learning. In our active learning paradigm, the training set contains both labeled and unlabeled examples. The learner

[1] These datasets cannot be used in our experiments, as they were originally created for the purpose of studying integrated quality of crowdsourced data rather than training learning models. *AC2* only provides URLs for workers to judge whether websites contain adult content. The websites include a mixture of text, images, and videos. *Trec10* only provides document IDs without their content. *SpamCF* is about judging whether an HIT is "spam." The objects are statistical tables and charts provided by Amazon Mechanical Turk. *Valence* only includes 100 news headlines, each of which contains several or dozens of words.

selects both labeled and unlabeled examples for additional repeated labels.

All traditional active learning methods involve evaluating the informativeness of unlabeled instances, which is called the instance selection (query) strategy [6]. Uncertainty sampling is a type of commonly used strategy, which is based on posterior probabilities, including margin sampling [18], entropy measure [20], multiple-instance measure [21], and least confidence measure [2]. Query by committee [22] is another commonly used type, where a committee of classifiers is used to evaluate unlabeled instances based on voting divergences. Besides these, there are some other strategies, such as expected model change [20], which selects the instance that would impart the greatest change to the current model; expected error reduction [9], [17], which selects the instance that would reduce the generalization error the most; variance reduction [11], which reduces the generalization error indirectly by minimizing output variance; and density-weighted methods [20], which focuses on the entire input space, avoiding querying outliers. The off-the-shelf strategies cannot be directly applied to multiple noisy labeling, because there are no ground truths provided by the labelers. In this paper, we define three novel uncertainty measures based on Bayesian estimation and a committee of learned models.

Following the pioneering work of repeated labeling [12], [24], Zhao *et al*. [33] employed crowdsourcing labelers in the active learning setting by incrementally relabeling the instances with a maximum loss function. Donmez and Carbonell [4] proposed proactive leaning, which focuses on selecting an optimal labeler as well as an optimal instance, at the same time using a decision theoretic approach. Based on modeling different levels of expertise among labelers [16], [29], Yan *et al*. [30] employed a probabilistic multilabeler model allowing learning from multiple annotators and provided an optimization formulation to select the most uncertain example to query its labels from annotators. To the best of our knowledge, none of the previous work considers the imbalanced distribution of the repeated labeling and deals with this issue in the active learning scenario.

III. Active Learning Framework

In this section, we first review the imbalanced multiple noisy labeling problem with its related annotations. Then, a novel active learning framework is proposed for crowdsourcing. The agnostic solution PLAT algorithm is introduced at the end.

A. Imbalanced Multiple Noisy Labeling

Considering a binary classification problem with samples $E = \{e_i\}_{i=1}^{N}$, every example contains an input feature portion x_i and an unknown true label y_i, denoted by $e_i = <x_i, y_i>$, $y_i \in \{-1, +1\}$. The labeler set of the system is denoted by $U = \{u_j\}_{j=1}^{R}$. Each example e_i is associated with a multiple noisy label set $\overrightarrow{l_i} = \{l_{ij}\}_{j=1}^{R}$ where the element l_{ij} comes from labeler j. All labels are shown by a matrix $L = \{\overrightarrow{l_i}\}_{i=1}^{N}$, $L \in \{-1, 0, +1\}^{N\times R}$, where 0 means that the labeler does not provide any label for this example. The column vector of L is notated as $\overrightarrow{c_j} = \{l_{ij}\}_{i=1}^{N}$. Every multiple noisy label set $\overrightarrow{l_i}$ contains $l_p^{(i)}$ positive labels and $l_n^{(i)}$ negative labels. The label of an example finally appearing in the training set and used for learning is induced from its multiple noisy label set. Formally, we have the following definitions.

Definition 1: Integrated label of an example i is the final label induced from its multiple noisy label set using a certain label integration method (MV, PLAT, etc.), notated as $\hat{y}_i$.

Definition 2: Integrated labeling quality of an example i is the probability that the integrated label is true, notated as $q_i = \Pr(\hat{y}_i = y_i | x_i, L)$.

Definition 3: Overall labeling quality of a labeler j is the probability that a label annotated by labeler j is true, notated as $p_j = \Pr(l_{ij} = y_i | E, \overrightarrow{c_j})$, which can be estimated as follows:

$$p_j = \frac{\sum_{i=1}^{N} \mathbf{I}\left(l_{ij}, y_i\right)}{N}, \text{ where } \mathrm{I}(x, y) = \begin{cases} 1, & x = y \\ 0, & x \neq y. \end{cases} \tag{1}$$

The imbalanced multiple noisy labeling issue is that the labelers exhibit different labeling qualities on different classes of the instances. Considering binary classification, a labeler j exhibits the labeling quality $p_P^{(j)}$ on positive examples and the labeling quality $p_N^{(j)}$ on negative examples, where

$$p_P^{(j)} = \Pr\left(l_{ij} = y_i | E, \overrightarrow{c_j}, y_i = +1\right) \tag{2}$$

$$p_N^{(j)} = \Pr\left(l_{ij} = y_i | E, \overrightarrow{c_j}, y_i = -1\right). \tag{3}$$

Without loss of generality, the imbalanced multiple noisy labeling means $p_N > p_P$. This requires us to assign as many examples as possible with positive labels with the premise that the overall accuracy stays at a relatively high level in the label integration procedure. Otherwise, a training set without enough positive examples would lead to a poor outcome for the learned model.

Here, we want to emphasize two different concepts again: imbalanced labeling and imbalanced dataset. The former is the object of this paper, which describes the behaviors of the labelers, and the latter describes an intrinsic property of a dataset. When imbalanced labeling meets an imbalanced dataset, the situation is even worse. The original scanty number of positive examples become fewer and fewer in the training set due to the imbalanced labeling.

B. Active Learning Framework

Fig. 1 describes the proposed active learning framework. The original training set $\mathcal{D}$ contains two parts: a labeled section $\mathcal{D}^{\mathcal{L}}$ and an unlabeled section $\mathcal{D}^{\mathcal{U}}$. The multiple noisy label set of an unlabeled example is empty. Then an instance selection strategy is used to select a group of instances with high uncertainty, which are the most ambiguous and need more labels to help determine which classes they belong to. After these instances are selected, multiple noisy labelers annotate these instances with certain correct probabilities. When an instance is given a nonempty multiple noisy label set, a label integration method (e.g., MV) is applied to induce an integrated label from its multiple label set. The examples with

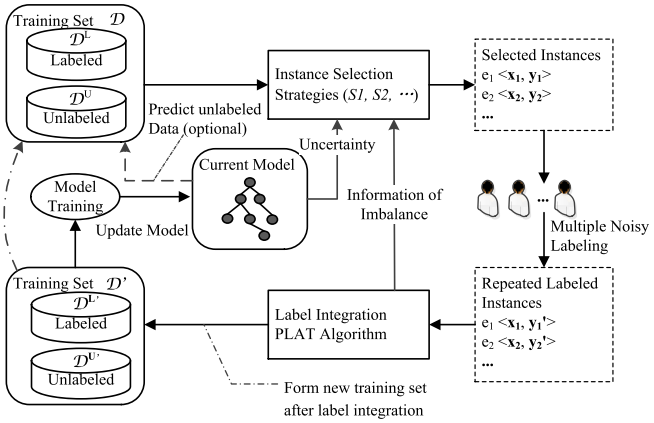


Fig. 1. Proposed active learning framework, which consists of four steps: instance selection, multiple noisy labeling, label integration, and model training.

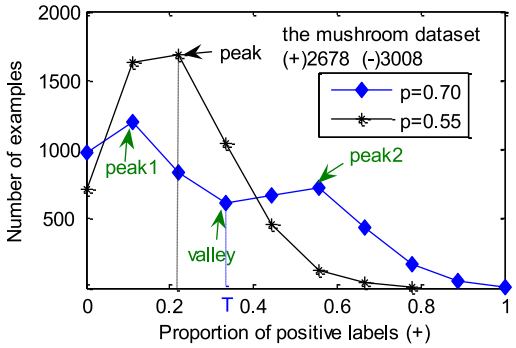


Fig. 2. Positive frequency distribution (PFD) curve which depicts the distribution of the number of examples with different proportions of positive labels under two different class labeling qualities *p*s [32].

integrated labels form a new training set $\mathcal{D}'$, and are used to train a model. The circle of updating the learned model does not stop until some conditions are met (e.g., the performance of the learned model meets a predefined requirement, it cannot be improved any more, or the budget of acquiring labels from labelers is exhausted). The learned model can also be used to predict the class labels of unlabeled instances in the training dataset, which is called semi-supervised learning [7]. Semi-supervised learning is out of the scope of this paper.

Besides replacing an oracle with multiple imperfect labelers in Fig. 1, there are two other main differences of our active learning framework from the traditional pool-based one [20], [26]. Firstly, in our active learning framework the instances selected to acquire more labels can be labeled ones that have nonempty multiple noisy label sets, or unlabeled ones. The integrated class label of a labeled example may change several times when it is selected to obtain more labels. Secondly, to simplify the learning process, the training set contains examples both with and without integrated labels. In traditional pool-based active learning, no unlabeled data are included in the training set.

C. Handle Imbalance During Label Integration

Many existing label integration methods [12], [16], [24], [27] focus on improving the accuracy of inferring the true labels of examples. They all hold the hypothesis that if an algorithm achieves higher accuracy of ground truth inference, it will form a better training set with integrated labels, and the final learned model is also better. This hypothesis holds only when labeling behaviors are balanced. Under an imbalanced labeling circumstance, a great number of negative labels will make examples with integrated negative labels overwhelm those with integrated positive labels. This will cause rare positive examples in the final training set. More seriously, if the underlying class distribution is also imbalanced, the training set does not even have any positive examples, which makes the subsequent training process completely impossible. The ground truth inference algorithm must have the ability of exploiting more positive examples, even allowing a tolerable sacrifice on the true negative accuracy rate to some extent. To achieve this goal, we utilize the heuristic algorithm PLAT [32] in the label integration procedure.

The basic principle of the PLAT algorithm is as follows. When using MV for label integration, as long as at least half of the labels in the multiple label set of an example are negative, we make a decision that the example is negative; otherwise it is positive, which means the decision boundary is 0.5 implicitly. However, when labeling is imbalanced, 0.5 shouldn't be treated as an appropriate decision boundary anymore. Supposing that labelers have the tendency to provide negative labels, the decision boundary should move toward 0 (< 0.5). Therefore, PLAT achieves the goal of increasing the number of positive examples in the training set by dynamically estimating a deterministic threshold T for label integration.

In PLAT, the threshold T is concretized as a certain frequency of positive labels (denoted by f^+) in a multiple label set. PLAT first calculates the f^+ value of each multiple label set, and then groups the examples with (almost) the same f^+ values together. Under imbalanced labeling, the labeling qualities of two classes (p_P and p_N) are different. Given an example with a true positive label, the number of positive labels (k) obeys a binomial distribution $b(k; R, p_P)$, where R is the size of the multiple noisy label set. Similarly, for a true negative example, the number of positive labels (k) also obeys a binomial distribution $b(k; R, 1\text{-}p_N)$. We plot the number of examples with the same f^+ value over the interval [0, 1] of all possible f^+ values. Then, we will get a positive frequency distribution (PFD) graph as Fig. 2 illustrates.

Fig. 2 illustrates two typical PFD curve shapes under different imbalance levels. The diamond-marked curve denotes the condition that the imbalance is not too serious and at this point the overall labeling quality may be relatively high (e.g., $p = 0.70$). We can find two distinct peaks on the curve. The left one designates the center of negative examples, since they all have small quantities of positive labels (i.e., f^+ values are very small). The right one designates the center of positive examples, since they all have larger quantities of positive labels (i.e., f^+ values are large). Between these two peaks there is a valley, which can be treated as the watershed of the two

classes. The best choice of an estimated T is the x-axis value of the valley. If imbalance is very prominent, at this point the overall labeling quality may be quite low (e.g., $p = 0.55$). The peaks will merge together, forming a curve like the asterisk-marked one in Fig. 2. Under this circumstance, no matter which class an example belongs to, it has a few positive labels and the f^+ values are indistinguishable. The only peak designates the point where negative and positive examples are mixed together to a maximal extent. The best choice of the estimated T is the x-axis value of this single peak. PLAT introduces a heuristic procedure EstimateThresholdPosition to estimate the optimal threshold T. EstimateThresholdPosition analyzes the distribution of the positive labels of all samples and obtains the estimation (denoted by t) of the threshold T.

After T is estimated as t, PLAT induces the integrated label from the multiple label set of each instance in the training set based on the threshold T. The examples with $f^+ > t$ are assigned with an integrated positive label. For those examples with $f^+ \leq t$, they may be assigned with integrated negative labels at a very high probability. During this procedure, PLAT tries to keep the ratio of the numbers of integrated positive and negative examples close to the true underlying class distribution of the training set.

The previous study [32] only investigates the performance of PLAT on ground-truth inference and traditional supervised learning. In this paper, we adapt it for active learning, where examples in a training set have varied sizes of multiple noisy label sets.

IV. Novel Instance Selection Strategies

Selecting informative instances to query is a key task of active learning. In crowdsourcing systems, instances are selected for which to acquire more noisy labels. We propose three novel selection strategies for our imbalanced labeling active learning framework proposed in the previous section.

A. Handling Imbalance During Instance Selection

Many instance selection strategies are based on uncertainty measures of examples. Under the imbalanced multiple noisy labeling scenario, the impact of labeling imbalance must be addressed. The level of uncertainty of an example is not only determined by some previous factors such as the ratio of different labels in the multiple label set and the probabilities that an example belongs to a certain class [12], [24], but also determined by the current level of imbalance of all the multiple noisy labels obtained for a dataset. That is, we cannot isolate the instance selection from the label integration. The instance selection should be completely integrated with the label integration.

Active learning is a dynamic procedure, and as we get more and more labels, the threshold T will be estimated time after time. These estimated ts provide information about the imbalance, which can be utilized in the computation of the uncertainty of an example during the instance selection procedure. Thus, considering label integration and instance selection as a whole, the level of imbalance as a factor of uncertainty calculation is the core of our methods.

B. Uncertainty Based on the Multiple Label Set and the Level of Labeling Imbalance

This strategy is proposed based on the assumption that the uncertainty of an example is only based on its multiple label set and the level of imbalance (abbreviated as MLSI).

We start our analysis from a balanced situation. Intuitively, if an example has the same number of two kinds of labels, it has the maximal uncertainty (the probability of being positive is 0.5). It seems that the frequency of the positive labels in a multiple label set can measure the level of uncertainly. However, this measure is inaccurate. Using frequency, the probabilities of being positive of two examples $e_1\{+, +, -, -, -\}$ and $e_2\{+, +, +, +, -, -, -, -, -, -\}$ are 0.4. However, they may have different levels of uncertainties, considering the numbers of labels. Since e_2 contains ten labels, it has a higher confidence of being positive. According to Bayesian estimation [8], we have the following proposition.

Proposition 1: Given an example with a multiple noisy dataset, which contains l_p positive and l_n negative labels, a posteriori probability of an integrated label's being the true label (i.e., the integrated labeling quality q as Definition 2 in Section III-A) obeys a Beta distribution with two parameters l_p+1 and l_n+1.

Proof: Since we do not know the true label y of an example, let's assume that the prior probability of its integrated labeling quality q follows the standard uniform distribution $q \sim U(0, 1)$. After l_p positive and l_n negative labels are observed, a posteriori probability of q can be calculated using the following Bayesian theorem:

$$\begin{aligned}\Pr(q|l_p, l_n) &= \frac{\Pr(q)\Pr(l_p, l_n|q)}{\Pr(l_p, l_n)} \\ &= \frac{C_{l_p+l_n}^{l_p} q^{l_p}(1-q)^{l_n}}{\int_0^1 C_{l_p+l_n}^{l_p} t^{l_p}(1-t)^{l_n} dt} \quad \left(= \frac{q^{l_p}(1-q)^{l_n}}{\int_0^1 t^{l_p}(1-t)^{l_n} dt}\right) \\ &= \frac{\Gamma(l_p+1+l_n+1+1)}{\Gamma(l_p+1)\Gamma(l_n+1)} q^{(l_p+1)-1}(1-q)^{(l_n+1)-1}\end{aligned}$$

where $\Gamma(n) = (n-1)!$. Let $\alpha = l_p$+1 and $\beta = l_n$+1, then $Pr(q\,|l_p, l_n)$ follows a Beta distribution, and has the probability density function as:

$$f(q; \alpha, \beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)} q^{\alpha-1}(1-q)^{\beta-1}. \tag{4}$$

Thus, $\Pr(q|l_p, l_n) \sim Beta(l_p + 1, l_n - 1)$. ■

For imbalanced labeling, the decision threshold t is dynamically calculated by the PLAT algorithm during label integration. Given a threshold t, we sum up all probabilities that the integrated label is the true label (i.e., the posterior probabilities of q) in the interval $[0, t]$, which is just the cumulative distribution function at the decision threshold of the Beta distribution

$$I_t(\alpha, \beta) = \sum_{j=\alpha}^{\alpha+\beta-1} C_{\alpha+\beta-1}^{j} t^j (1-t)^{\alpha+\beta-1-j}. \tag{5}$$

If this summation of the posterior probabilities of q and its complement are the same, it means the overall probability of the integrated label to be the true label in the interval $[1, t]$ is

0.5 (i.e., $I_t(\alpha, \beta) = 1 - I_t(\alpha, \beta) = 0.5$), which is the most uncertain case. Thus, the smaller the values of $I_t(\alpha, \beta)$ or $1 - I_t(\alpha, \beta)$, the smaller the uncertainty level will be. Since we do not know the true label of the example, we define the uncertainty measure as the tail probability below the decision threshold t of the Beta distribution, which is

$$S_{\text{MLSI}} = \min\{I_t(l_p + 1, l_n + 1),\ 1 - I_t(l_p + 1, l_n + 1)\}. \quad (6)$$

When acquiring additional labels, the MLSI strategy always selects the example I^*_{MLSI} with the largest S_{MLSI} value (the most uncertain instance)

$$I^*_{\text{MLSI}} = \arg\max_i \left\{ S^{(i)}_{\text{MLSI}}, 1 \le i \le N \right\}. \quad (7)$$

C. Uncertainty Based on Class Membership Probabilities in the Learned Model and the Level of Labeling Imbalance

MLSI assumes that different examples are labeled independently. The labels assigned to one example do not have any association with the labels of other examples. Unlike MLSI, using the multiple noisy label set of the example when measuring uncertainty, this strategy utilizes only the estimated class membership probability provided by a current learned model, as well as the level of labeling imbalance (abbreviated as CMPI). CMPI considers the dependency of the labeled and unlabeled examples, and assumes that similar examples will be labeled similarly. Whether an example should acquire additional labels depends on the uncertainty value provided by the learners.

In order to avoid the performance variance of the current immature learned model, CMPI builds a set of learned models H_i ($1 \le i \le m$, m being the number of learned models), and uses these models to predict the class membership of an example. The probability that an example is classified as positive can be calculated as follows:

$$S_p = \frac{1}{m} \sum_{i=1}^{m} \Pr(+|x, H_i) \quad (8)$$

where $Pr(+|x, H_i)$ is the probability of classifying the example x as positive by the learned model H_i. In our experiments (Section V), we set $m = 10$, and the models are trained using the random forest [1], which usually has a more stable performance compared with a single decision tree.

The uncertainty measure under balanced circumstances (threshold $t = 0.5$) can be easily defined as the distance between the estimated probability of being positive or negative and the threshold 0.5. An example with a larger distance has a lower level of uncertainty

$$S_{\text{CMPI}} = 0.5 - \left| S_p - 0.5 \right|. \quad (9)$$

If $S_p < 0.5$, S_{CMPI} is S_p and otherwise S_{CMPI} is 1-S_p. Under imbalanced labeling circumstances, we cannot simply replace 0.5 with the decision boundary t in (9). When the threshold t moves toward 0, the scales of the interval $[0, t]$ and $[t, 1]$ are different. When S_p is given by the learners, the uncertainty S_{CMPI} must be calculated according to its relative position in ranges $[0, t]$ or $[t, 1]$ rather than its absolute value. This means we must tune the uncertainty to the same scale 0.5. Now, S_{CMPI} can be calculated as follows:

$$S_{\text{CMPI}} = \begin{cases} S_p \cdot \frac{0.5}{t} & S_p < t \\ (1 - S_p) \cdot \frac{0.5}{1-t} & S_p \ge t. \end{cases} \quad (10)$$

When acquiring additional labels, the CMPI strategy always selects the example $I_{\text{CMPI}}{}^*$ with the largest S_{CMPI} value

$$I^*_{\text{CMPI}} = \arg\max_i \left\{ S^{(i)}_{\text{CMPI}}, 1 \le i \le N \right\}. \quad (11)$$

D. Hybrid Uncertainty

MLSI only utilizes the multiple label set of each example in a dataset independently, whereas CMPI considers the dependency of examples in the dataset and only utilizes the probabilities estimated by the learners. Thus, CMPI has complementary characteristics to MLSI, and vice versa. A hybrid strategy based on these compound factors and the level of labeling imbalance (denoted as CFI) directly integrates two uncertainties S_{MLSI} and S_{CMPI} together. There are different ways to integrate two uncertainty measures. We tried the arithmetic mean and the geometric mean, and found that the geometric mean performs better than the arithmetic mean. The integrated measure is as follows:

$$S_{\text{CFI}} = \sqrt{S_{\text{MLSI}} \cdot S_{\text{CMPI}}}. \quad (12)$$

When acquiring additional labels, the CFI strategy always selects the example $I_{\text{CFI}}{}^*$ with the largest S_{CFI} value

$$I^*_{\text{CFI}} = \arg\max_i \left\{ S^{(i)}_{\text{CFI}}, 1 \le i \le N \right\}. \quad (13)$$

As we mentioned above, these strategies are not only suitable for imbalanced labeling circumstances, but also suitable for balanced circumstances. Under balanced circumstances, PLAT will estimate the threshold at around 0.5 which is the same as the threshold used by MV. All these strategies are backward compatible with the strategies in previous work [12], [24].

V. Experiments

In this section, we will first conduct experiments to empirically verify the performance of our proposed active learning framework with our three instance selection strategies MLSI, CMPI, and CFI on 12 UCI datasets with synthetic multiple noisy labels. Then, we will apply our methods to a real-world dataset with noisy labels collected from Amazon Mechanical Turk to further investigate and analyze their performance in a real-world scenario.

A. Experiments Using Synthetic Multiple Noisy Label Sets

We first conduct experiments on 12 datasets from the UCI database repository,[2] shown in Table II. These datasets are chosen because they have different underlying class distributions, different numbers of examples, and different numbers

[2] *page-block0* and *kddcup-ivb* (also named *kddcup-rootkit-imap_vs_back*) are available at http://sci2s.ugr.es/keel/imbalanced.php. These datasets are derived from the UCI database repository. We still call them UCI datasets for consistency and convenience.

TABLE II
UCI DATASETS USED IN EXPERIMENTS

Dataset	#Attributes	#Examples	#Pos	*d*
mushroom	23	8124	3916	0.482
kr-vs-kp	37	3196	1527	0.478
musk(clean1)	169	476	207	0.435
spambase	58	4601	1813	0.394
tic-tac-toe	10	958	332	0.347
waveform	41	5000	1692	0.338
abalone	9	4177	882	0.211
bankmarket	16	4521	521	0.115
page-block0	10	5472	559	0.102
thyroid	30	3772	291	0.077
car-eval	6	1728	65	0.038
kddcup-ivb	41	2225	23	0.010

Each example in these twelve UCI datasets is assigned a synthetic multiple noisy label set. The initial state of each label set is empty.

of attributes. When necessary, we convert multiple classes into binary classes. Specifically, for the *waveform* dataset, we integrate classes 1 and 2; for *abalone*, we keep age in range [6–12] as negative and the others as positive; for *thyroid*, we keep the negative class and integrate the other three classes into a positive class; for *car-eval*, we treat "*vgood*" as positive and integrate the other three classes to form the negative class. The parameter d in Table II is the proportion of true positive examples in each dataset.

For each UCI dataset, 30% of examples are held out, in every run, as the test set, and the rest (70%) are used as training examples with simulated multiple noisy labels. Each experiment is repeated ten times with a different random data partition, and their average results are reported. Each average result is presented in the form of a learning curve with six standard deviation values at different stages of the active learning procedure.

To simulate multiple noisy labels, we first hide the labels of training examples. Then we create ten virtual labelers with different labeling qualities (the differences are controlled by the seeds used for generating Gaussian distributions). Each labeler has two quality parameters p_P and $p_N \cdot p_P$ and p_N are extracted from two Gaussian distributions $N(0.4, 0.15^2)$ and $N(0.8, 0.15^2)$, respectively.[3] When a label is acquired for an example in the experiment, we randomly choose a labeler and generate a label according to its gold label y. If the example already has multiple labels, two labels are assigned to the selected example; otherwise, one label is assigned to this example. This will make the number of labels odd so that MV can always determine a majority. After 1% of examples in the training set have obtained new labels, we use PLAT to infer the integrated label for every example with a nonempty multiple label set, and then use J48, the enhanced implementation of C4.5 [15] in WEKA [28], to induce a classifier.

[3]These two normal distributions are based on our investigation on some real-world datasets. When imbalanced labeling exists, p_P often falls into the interval [0.25∼0.55] and has a high probability of around 0.4, and p_N often falls into [0.65∼0.95] and has a high probability of around 0.8. The analysis of these datasets is available at http://sun0.cs.uca.edu/∼ssheng/DA/pdf/biasedcrowd.pdf.

We use the area under the ROC curve (AUC) as the performance metric [5], [14], because some datasets (with $d < 0.4$) have imbalanced underlying class distributions, and also because imbalanced labeling will make the negative examples outnumber the positive ones. AUC can utilize the probabilistic outputs of the induced classifier and provide a more comprehensive assessment [10].

B. Effectiveness of the Proposed Methods

To the best of our knowledge, there are no other studies on active learning with the imbalanced multiple noisy labeling. We compare our proposed strategies with the most similar ones proposed in [24]. In their work, three active learning strategies LU, MU, and LMU were proposed. These strategies are similar to ours: LU is only based on the multiple label set of each example; MU is only based on the current learned model; and LMU combines the uncertainty measures of LU and MU together. Note that LU, MU, and LMU are all instance selection strategies. All of them were proposed with MV as the label integration method [24]. Another better label integration method MVBeta, which considers the weights of different labels in a multiple label set, was proposed after [23], which can be directly applied to LU, MU, and LMU strategies, forming LUβ, Muβ, and LMUβ methods. We compare our MLSI with the corresponding LU and LUβ, our CMPI with the corresponding MU and MUβ, and our CFI with the corresponding LMU and LMUβ, respectively, in our experiments.

Fig. 3 shows the comparison results on the *mushroom*, *musk*, *tic-tac-toe*, *abalone*, *page-block0*, and *car-eval* datasets (i.e., datasets in odd-numbered lines in Table II). These datasets have different underlying class distributions (i.e., the different d parameters shown in Table II). The d values are in decreasing order. The results on these six datasets are consistent and obvious. Our proposed MLSI method is conspicuously superior to its counterparts LU and LUβ, and our CMPI and CFI methods are also conspicuously superior to their counterparts (i.e., MU and MUβ, LMU and LMUβ) respectively. It is easy to draw the following conclusions.

1) Under the imbalanced labeling circumstance, active learning works only if adequate positive examples are correctly inferred. The previous active learning strategies LU, MU, LMU based on MV and their variants (i.e., LUβ, MUβ, and LMUβ based on MVBeta) do not work at all, because their integration methods (MV and MVBeta) do not provide adequate attention to the potential positive examples when inferring the integrated labels of examples. However, PLAT works. Our empirical study shows that our three strategies based on PLAT perform very well. This is because PLAT is used in their label integration procedures. The performance comparison among our three strategies will be discussed in the next section.
2) The comparison results provide convincing evidence that PLAT is also suitable for datasets where examples have variable numbers of multiple noisy labels. Theoretically, PLAT only concerns the frequency of positive repeated

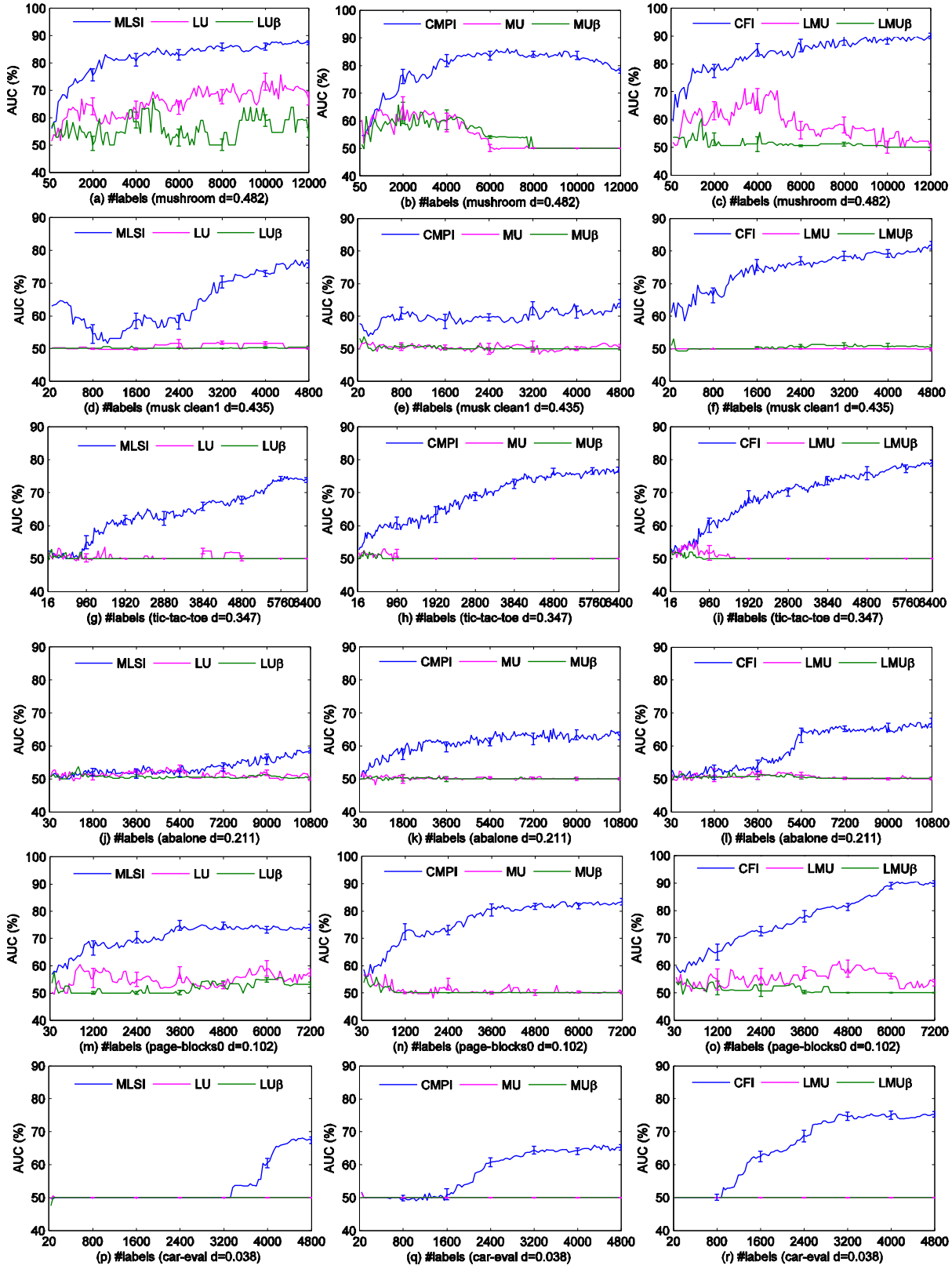


Fig. 3. Comparisons of our proposed strategies MLSI, CMPI, CFI with LU, MU, LMU and their variants LUβ, MUβ, LMUβ on six UCI datasets with synthetic multiple noisy labeling sets. (a), (d), (g), (j), (m), and (p) are for the comparisons of MLSI with LU and its variant LUβ. (b), (e), (h), (k), (n), and (q) are for the comparisons of CMPI with MU and its variant MUβ. (c), (f), (i), (l), (o), and (r) are for the comparisons of CFI with LMU and its variant LMUβ.

labels of each example, regardless of the size of its multiple label set. This feature has not been confirmed until our active learning framework is introduced.

3) The underlying class distribution has obvious impact on the performance of active learning. On the datasets whose underlying class distributions are nearly balanced,

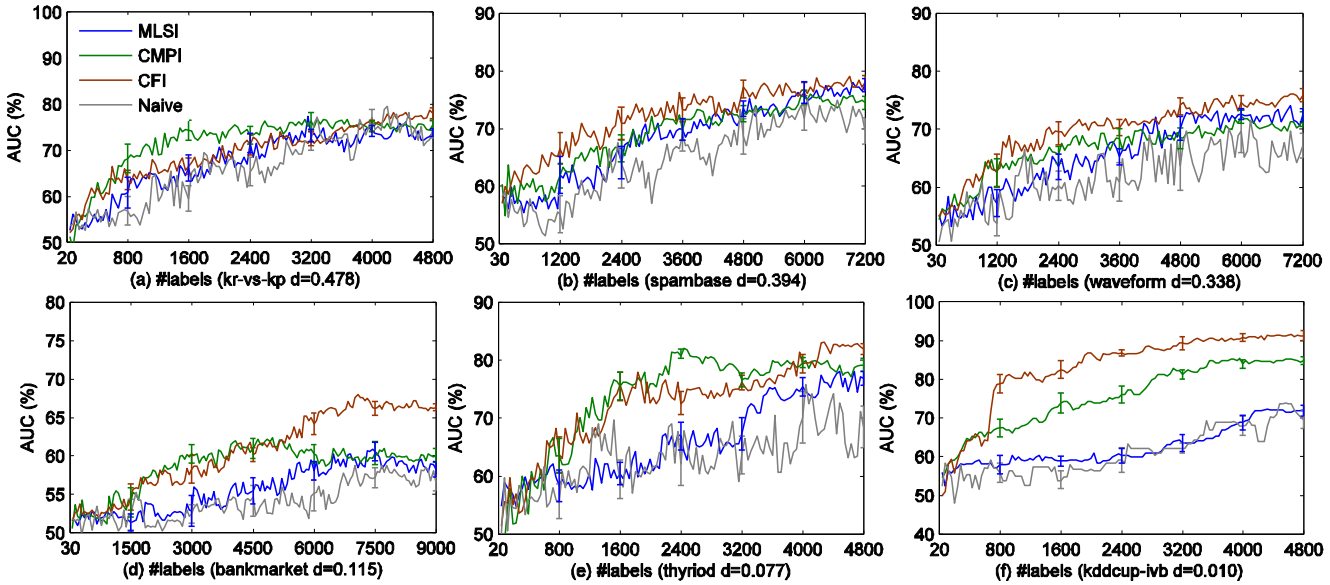


Fig. 4. Comparisons among MLSI, CMPI, CFI, and the Naive instance selection strategy on six UCI datasets with synthetic multiple noisy labeling sets. (a) Comparison on the dataset *kr-vs-kp*. (b) Comparison on the dataset spambase. (c) Comparison on the dataset *waveform*. (d) Comparison on the dataset *bankmarket*. (e) Comparison on the dataset *thyroid*. (f) Comparison on the dataset *kddcup-ivb.*

such as *mushroom* (d = 0.482, which means the numbers of positive and negative examples are almost the same), it is easier for the active learning strategies to achieve high performance. On this dataset even LU shows improvement on active learning to some extent. On the contrary, some imbalanced datasets make the performance curve a gentle ascendant (e.g., *abalone* and *car-eval*). Neither MV nor MVBeta can work on imbalanced datasets with completely imbalanced labeling. Our proposed methods are less affected by this factor, since our methods have considered the imbalance coming from both the underlying class distribution and the labeling behaviors.

4) Our methods have reasonable standard deviation in the interval [± 0.3%, ±3%], and the standard deviation at the early stage is a bit larger than that at the later stationary stage when the performance fluctuates slightly. In Figs. 3 and 4, we show six standard deviation bars at different stages of active learning.

We also conduct comparisons on the other six datasets *kr-vs-kp*, *spambase*, *waveform*, *bankmarket*, *thyroid,* and *kddcup-ivb.* Although the six datasets have different underlying class distributions, and their underlying class distributions are different from the previous six datasets (*mushroom*, *musk*, *tic-tac-toe*, *abalone*, *page-blocks0,* and *car-eval*) shown in Table II, the experimental results on these six datasets are similar to those on the previous six datasets, and the conclusions drawn from the experimental results of the previous six datasets are sustainable. Thus, we do not show the experimental results of these six datasets in Fig. 3. We will show the experimental comparisons on these six datasets in the next section.

C. Comparisons Among the Proposed Instance Selection Strategies

We also compare the performance of the proposed three novel instance selection strategies MLSI, CMPI, and CFI. In our experiments we include another strategy for comparison, named the Naive strategy, which does not consider the imbalance information and simply chooses the examples with fewest labels in each iteration. Fig. 4 only shows the experimental results on the *kr-vs-kp*, *spambase*, *waveform*, *bankmarket*, *thyroid*, and *kddcup-ivb* datasets. From Figs. 3 and 4, we can draw conclusions for our proposed three instance selection strategies as follows.

1) The three proposed instance selection strategies improve the performance of active learning compared to the Naive strategy. The curve of the Naive strategy still shows a little improvement on the balanced (e.g., *kr-vs-kp*) or moderately imbalanced (e.g., *spambase*) datasets, but it no longer shows improvement on the datasets with a higher level of imbalance (e.g., *waveform*, *bankmarket*, and *thyroid*).
2) The learning curve of the Naive strategy has a very large variance. The reason is that this strategy blindly obtains more labels for the instance with the fewest labels, regardless of whether its true label is easy to induce. The consequence is that this manner may periodically increase the uncertainty of the instance, even when its uncertainty is relatively the lowest. This directly downgrades the performance of the learned model. Our proposed strategies overcome this defect, so they have smoother active learning curves.
3) The shapes of the learning curves of MLSI and CMPI are quite different. The learning curve of CMPI quickly ascends at the beginning, when the multiple noisy label set of each example contains a small number of labels. After that, the curve gradually becomes flat. This feature of CMPI is very conspicuous in Figs. 3(b), (e), (k), (n) and (q) and 4(a) and (c)–(f). In most cases, the curve of MLSI keeps ascending until it reaches its maximum performance. Therefore, at the early stage of active learning, CMPI has better performance than MLSI. The reason

TABLE III
MEAN VALUES OF AUCS ON 12 UCI DATASETS

Dataset	MLSI	CMPI	CFI
mushroom	81.9±1.9	*79.2±1.4*	**83.8±1.8**
kr-vs-kp	68.6±2.1	**71.8±1.7**	*67.5±2.0*
musk(clean1)	64.1±2.6	*59.7±1.9*	**74.4±1.8**
spambase	*68.3±1.8*	69.1±1.9	**70.5±1.7**
tic-tac-toe	*63.7±1.6*	66.8±2.1	**68.0±1.8**
waveform	*65.8±1.9*	66.8±1.8	**68.7±1.8**
abalone	*53.4±2.0*	**60.9±1.7**	59.6±1.9
bankmarket	*55.7±1.5*	58.7±1.4	**59.5±1.5**
page-block0	*70.5±1.6*	76.4±1.7	**76.9±2.1**
thyroid	*66.0±2.7*	**73.7±2.0**	72.6±2.5
car-eval	*55.8±1.4*	57.8±1.3	**65.6±1.4**
kddcup-ivb	*62.4±2.3*	75.4±1.6	**83.2±1.8**

TABLE IV
STUDENT-t TEST RESULTS (p-VALUE = 0.05)

Algorithm	MLSI	CMPI
CMPI	7/2/2	—
CFI	9/3/0	4/7/1

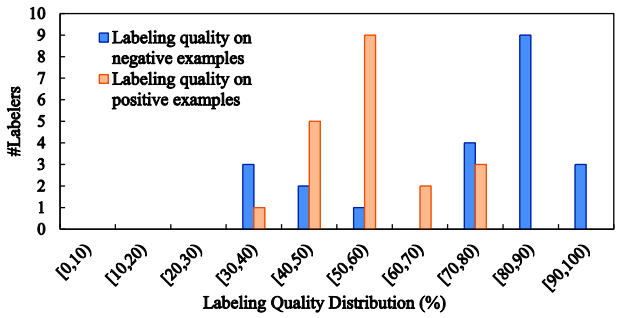


Fig. 5. Labeling quality distribution of all labels on different classes. About 80% of labelers have a labeling quality on the negative examples in the range [70, 100], and about 70% of labelers have a labeling quality on the positive examples in the range [40, 60].

is that MLSI relies on the distribution of the different labels in the multiple noisy label set of each instance. At the beginning, the multiple label set of each instance contains a few labels, which is inadequate for label estimation and integration. Thus, MLSI takes a long start-up time to show performance improvement, especially under adversarial circumstances, because adversarial labelers increase the randomness of the whole multiple label sets by assigning opposite labels at a high frequency. When there are finally enough labels collected, MLSI begins to make better estimations. CMPI is also impacted by the sizes of the multiple label sets at the early stage of active learning. However, the size of the impact on CMPI is much less than that on MLSI. This is because CMPI mainly relies on the relationship among instances. In most cases, the eventual performance of MLSI can be almost the same as that of CMPI [refer to Fig. 3(a), (b), (d), (e), (g), (h), (p), and (q) and all of Fig. 4 expect for Fig. 4(f)]. However, in some cases, the eventual performance of MLSI is inferior to that of CMPI [refer to Figs. 3(j), (k), (m), and (n) and 4(f)].

4) When the active learning procedure arrives at its stationary stage, the performance of the learned model can no longer be improved significantly. At this point, the hybrid strategy CFI exhibits the best performance. Experimental results on all datasets show that: 1) at the early stage, the performance of CFI may be inferior to that of CMPI but is certainly better than that of MSLI and 2) at the later stationary stage, the performance of CFI is definitely superior to that of MSLI and CMPI. At the stationary stage, on two moderately imbalanced datasets (*muskclean1*, *waveform*) and five extremely imbalanced datasets (*abalone*, *bankmarket*, *page-blocks0*, *car-eval,* and *kddcup-ivb*), CFI obviously outperforms the other two strategies, and on the other datasets CFI is not obviously inferior to them.

Although CFI has the best performance in the later stationary stage of active learning, active learning is a dynamic procedure in which the performance of a model is gradually improved. In order to investigate the overall performance, it is better to use the mean value of all AUCs on the learning curve. Table III shows the mean values of AUCs after running the algorithms ten times on every UCI dataset.

Table III shows that in most cases CFI has the best mean values. We conduct a student-t test using the results on each dataset. We set statistical significance to 95% (i.e., p-value = 0.05). Table IV shows the student-t test results in the form of "*win/tie/loss*" on 12 UCI datasets. Obviously, MLSI has the worst performance since it seldom wins. Comparing with CMPI, CFI wins on four datasets, ties on seven datasets, and loses on one dataset.

D. Performance on Real-World Crowdsourcing Dataset

In order to investigate the performance of our proposed methods and to study whether they can be applied to actual crowdsourcing systems, we conduct the following experiment. We use the *adult* dataset in UCI which contains 32 561 instances. This dataset's described purpose is to predict whether a person's income exceeds \$50K/year based on the 1994 census data. The target concept contains two classes $\leq$ 50K (negative) and $>$ 50K (positive). We extract 600 instances (300 positive and 300 negative) as a training set and another 300 instances (150 positive and 150 negative) as a test set. We hide the true labels of all examples in the training set. For each example, we create a human intelligence task (HIT) on Amazon Mechanical Turk, which requires anonymous workers to label it according to its 14 attributes.

Since our proposed instance selection strategies may acquire labels for one example many times, we require that every HIT must be completed by 20 workers to ensure that we have enough labels available during the active learning procedure. That is, every instance obtains 20 labels from different labelers.

We also ask every labeler to label at least 100 examples. We collect a total of 12 000 labels from Amazon Mechanical Turk

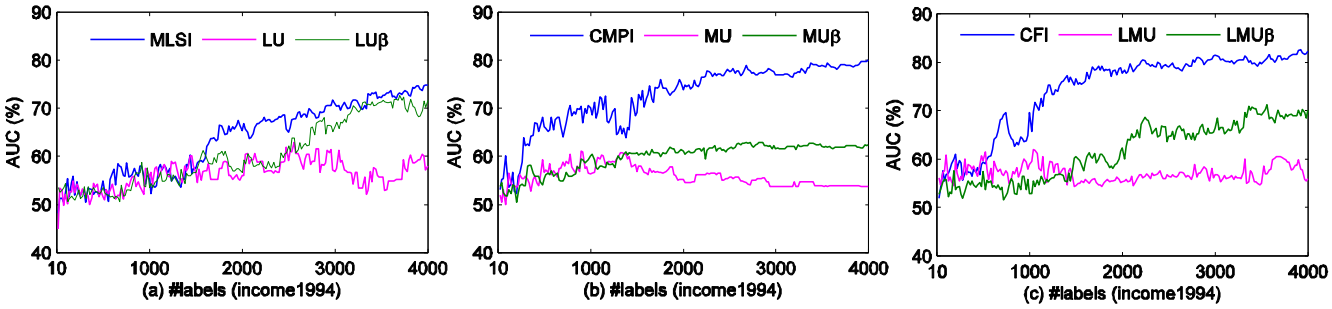


Fig. 6. Comparison with LU, MU, LMU and their variants LUβ, MUβ, LMUβ on the real-world dataset *income1994*.

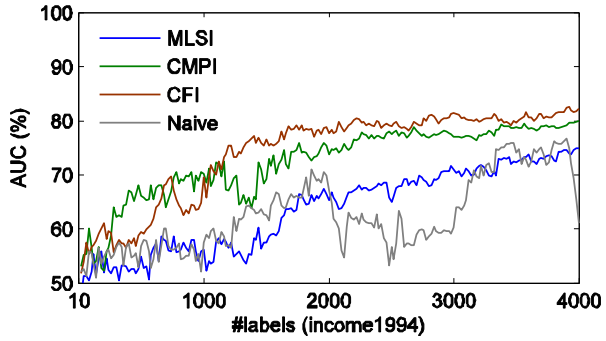


Fig. 7. Comparison among MLSI, CMPI, CFI, and the Naive instance selection strategy on the real-world dataset *income1994*.

and form the multiple noisy label dataset named *income1994*. Therefore, every instance in the training set has a noisy label pool containing 20 collected labels. During active learning, when the learner acquires an additional label for an example, it randomly chooses a label from its noisy label pool of this example, and adds this label into its multiple label set. Each label in the noisy label pool can be chosen only once.

We analyze this dataset and find that most labelers prefer to give a conservative prediction that the income of a person is equal to or less than ($\leq$) \$50K/year (negative class). (The positive class represents that a person's income is higher than \$50K/year.) Since we have the true labels of the training examples, we calculate the accuracies on true negative and true positive examples for each labeler respectively to estimate his/her p_N and p_P. The distribution of p_Ps and p_Ns of 20 labelers is shown in Fig. 5. We can infer that the labeling qualities on the two classes are quite different. For negative examples, the correct rates of labels are concentrated in the range 70%$\sim$90%, and for positive ones, they are concentrated in the range 40%$\sim$60%. Thus, *income1994* is a typical imbalanced multiple noisy labeling dataset.

We conduct active learning on this dataset to compare our proposed three methods with LU, MU, LMU and its variants LUβ, MUβ, and LMUβ respectively. Fig. 6 shows the comparative results. We also show the comparative results among our proposed three methods together with the naive strategy in Fig. 7.

From Figs. 6 and 7 we can draw the following conclusions.

1) Our proposed methods are all superior to their respective counterparts. Compared to Fig. 3, the learning curves in this experiment appear to contain a slightly larger variance. This is because the labeling quality in the real-world crowdsourcing system is more complex than that in simulations where p_P and p_N are extracted from some normal distributions.
2) The conclusions drawn from Fig. 3 are sustainable. At the early stage of active learning, CMPI has the best performance, and at the later stage the performance of CFI is obviously superior to that of CMPI and MLSI. The naive strategy has the worst performance.
3) Since there are nine labelers whose labeling qualities on positive examples are in the range 50%$\sim$60% (shown in Fig. 5) in the *income1994* dataset, the total quality on positive examples is a little better than those of our simulations. Thus, we find that LUβ, Muβ, and LMUβ perform better on this real-world dataset (refer to Fig. 6), which proves that MVBeta is superior to MV when the labeling behavior is balanced or slightly imbalanced. However, our proposed methods are still significantly superior to LUβ, Muβ, and LMUβ, respectively on this dataset.

VI. Conclusion

Many crowdsourcing systems integrate active learning to improve learning performance when labelers are imperfect. Facing the imbalanced multiple noisy labeling issue, the traditional active learning framework encounters great challenges in label integration and instance selection. This paper proposed a novel active learning framework where both labeled and unlabeled instances can be selected to obtain more labels. When a certain number of instances obtain additional labels, the framework conducts label integration to induce the integrated label from the multiple noisy label set of each example in the training set. After the training set has been updated with new integrated labels, the framework uses this set to update the learned model.

In order to handle the imbalanced multiple noisy labeling issue, in the label integration procedure, we introduced the PLAT algorithm as a new active learning instance selection algorithm. The experimental results on datasets with different underlying class distributions showed that introducing the PLAT algorithm in the label integration phase is a critical factor to the success of active learning. Because only PLAT can give the minority (positive) class special treatment during label integration, the training set contains enough

minority examples for sequential supervised learning. The other label integration methods such as MV and its variant MVBeta do not have this feature, so they cannot work well in active learning with imbalanced labeling. In order to improve the performance of active learning, we proposed three novel instance selection strategies MLSI, CMPI, and CFI. These strategies are not only based on some uncertainty measures, but also take advantage of dynamic information about imbalance derived from label integration using the PLAT algorithm. Among these strategies, CFI combines the advantages of MLSI and CMPI, and so has the highest performance.

We conducted experiments on 12 datasets with synthetic multiple noisy labels, and also applied our methods to a real-world dataset with labels collected from Amazon Mechanical Turk. All experimental results demonstrated that our proposed active learning framework achieved very high performance.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their insightful and constructive comments and suggestions that have helped improve the quality of this paper.

REFERENCES

[1] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
[2] A. Culotta and A. McCallum, "Reducing labeling effort for structured prediction tasks," in *Proc. 20th Nat. Conf. Artif. Intell.*, 2005, pp. 746–751.
[3] S. Dasgupta, D. Hsu, and C. Monteleoni, "A general agnostic active learning algorithm," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 20. 2008, pp. 353–360.
[4] P. Donmez and J. G. Carbonell, "Proactive learning: Cost-sensitive active learning with multiple imperfect oracles," in *Proc. ACM Conf. Inf. Knowl. Manage. (CIKM)*, 2008, pp. 619–628.
[5] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, 2006.
[6] Y. Fu, X. Zhu, and B. Li, "A survey on instance selection for active learning," *Knowl. Inf. Syst.*, vol. 35, no. 2, pp. 249–283, 2013.
[7] A. Fujino, N. Ueda, and M. Nagata, "Adaptive semi-supervised learning on labeled and unlabeled data with different distributions," *Knowl. Inf. Syst.*, vol. 37, no. 1, pp. 129–154, 2013.
[8] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian Data Analysis*. Boca Raton, FL, USA: Chapman and Hall, 2004.
[9] Y. Guo and R. Greiner, "Optimistic active learning using mutual information," in *Proc. Int. Joint Conf. Artif. Intell. (IJCAI)*, 2007, pp. 823–829.
[10] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263–1284, Sep. 2009.
[11] S. C. H. Hoi, R. Jin, and M. R. Lyu, "Large-scale text categorization by batch mode active learning," in *Proc. Int. Conf. World Wide Web (WWW)*, Edinburgh, U.K., 2006, pp. 633–642.
[12] P. G. Ipeirotis, F. Provost, V. S. Sheng, and J. Wang, "Repeated labeling using multiple noisy labelers," *Data Mining Knowl. Discov.*, vol. 28, no. 2, pp. 402–441, 2014.
[13] C. X. Ling and J. Du, "Active learning with direct query construction," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining (KDD)*, Las Vegas, NV, USA, 2008, pp. 480–487.
[14] C. Parker, "On measuring the performance of binary classifiers," *Knowl. Inf. Syst.*, vol. 35, no. 1, pp. 131–152, 2013.
[15] J. R. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo, CA, USA: Morgan Kaufmann, 1992.
[16] V. C. Raykar *et al.*, "Learning from crowds," *J. Mach. Learn. Res.*, vol. 11, pp. 1297–1322, Apr. 2010.
[17] N. Roy and A. McCallum, "Toward optimal active learning through sampling estimation of error reduction," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2001, pp. 441–448.
[18] T. Scheffer, C. Decomain, and S. Wrobel, "Active hidden Markov models for information extraction," in *Proc. Int. Conf. Adv. Intell. Data Anal. (CAIDA)*, 2001, pp. 309–318.
[19] B. Settles. (2009). "Active learning literature survey," Dept. Comput. Sci. Univ. Wisconsin–Madison, Madison, WI, USA, Comput. Sci. Tech. Rep. 1648 [Online]. Available: http://research.cs.wisc.edu/techreports/2009/TR1648.pdf
[20] B. Settles and M. Craven, "An analysis of active learning strategies for sequence labeling tasks," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Oct. 2008, pp. 1070–1079.
[21] B. Settles, M. Craven, and S. Ray, "Multiple-instance active learning," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS),* vol. 20. 2008, pp. 1289–1296.
[22] H. S. Seung, M. Opper, and H. Sompolinsky, "Query by committee," in *Proc. ACM Workshop Comput. Learn. Theory*, 1992, pp. 287–294.
[23] V. S. Sheng, "Simple multiple noisy label utilization strategies," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, 2011, pp. 635–644.
[24] V. S. Sheng, F. Provost, and P. Ipeirotis, "Get another label? Improving data quality and data mining using multiple, noisy labeler," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining (KDD)*, Las Vegas, NV, USA, 2008, pp. 614–662.
[25] A. Sheshadri and M. Lease, "SQUARE: A benchmark for research on computing crowd consensus," in *Proc. 1st AAAI Conf. Human Comput. Crowdsourcing (HCOMP)*, 2013, pp. 156–164.
[26] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2000, pp. 999–1006.
[27] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, and J. Movella, "Whose vote should count more: Optimal integration of labels from labelers of unknown expertise," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS),* vol. 22. 2009, pp. 2035–2043.
[28] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed. San Mateo, CA, USA: Morgan Kaufman, 2005.
[29] Y. Yan, R. Rosales, G. Fung, and J. G. Dy, "Modeling multiple annotator expertise in the semi-supervised learning scenario," in *Proc. Conf. Uncertainty Artif. Intell. (UAI)*, 2010, pp. 674–682.
[30] Y. Yan, R. Rosales, G. Fung, and J. G. Dy, "Active learning from crowds," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, Bellevue, WA, USA, 2011, pp. 1161–1168.
[31] J. Zhang, X. Wu, and V. S. Sheng, "Imbalanced multiple noisy labeling for supervised learning," in *Proc. 27th AAAI Conf. Artif. Intell. (AAAI)*, 2013, pp. 1651–1652.
[32] J. Zhang, X. Wu, and V. S. Sheng, "A threshold method for imbalanced multiple noisy labeling," in *Proc. 2013 IEEE/ACM Int. Conf. Adv. Soc. Netw. Anal. Mining (ASONAM)*, pp. 61–65.
[33] L. Zhao, G. Sukthankar, and R. Sukthankar, "Incremental relabeling for active learning with noisy crowdsourced annotations," in *Proc. IEEE 3rd Int. Conf. Soc. Comput. (ICSC)*, 2011, pp. 728–733.
[34] X. Zhu, P. Zhang, Y. Shi, and X. Lin, "Active learning from stream data using optimal weight classifier ensemble," *IEEE Trans. Syst., Man, Cybern. B, Cybern.,* vol. 40, no. 6, pp. 1607–1621, Dec. 2010.

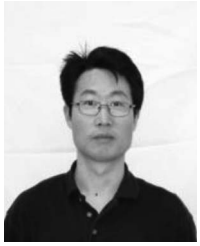
Jing Zhang (S'14) received the B.S. degree in engineering from Anhui University, Hefei, China, in 2003, and the M.S. degree in computer science from the Graduate University of Chinese Academy of Sciences, Beijing, China, in 2006. He is currently pursuing the Ph.D. degree in computer science from the Hefei University of Technology, Hefei.

His current research interests include data mining, machine learning, and their applications in business and industry. He served as an External Reviewer for several international conferences and workshops.

Xindong Wu (SM'95–F'11) received the bachelor's and master's degrees in computer science from the Hefei University of Technology, Hefei, China, and the Ph.D. degree in artificial intelligence from the University of Edinburgh, Edinburgh, U.K.

He is a Yangtze River Scholar at the School of Computer Science and Information Engineering, Hefei University of Technology, a Professor of Computer Science at the University of Vermont, Burlington, VT, USA, and a Fellow of the AAAS. His current research interests include data mining, knowledge-based systems, and web information exploration.

Dr. Wu is the Steering Committee Chair of the IEEE International Conference on Data Mining (ICDM), the Editor-in-Chief of the *Knowledge and Information Systems* (Springer), and an Editor-in-Chief of the Springer Book Series on *Advanced Information and Knowledge Processing*. He was the Editor-in-Chief of the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING (IEEE Computer Society) between 2005 and 2008. He served as a Program Committee Chair/Co-Chair for ICDM'03, the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining in 2007, the 19th ACM Conference on Information and Knowledge Management in 2010, and the 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining.



Victor S. Sheng (M'11) received the master's degree in computer science from the University of New Brunswick, Fredericton, NB, Canada, in 2003, and the Ph.D. degree in computer science from the Western University, London, ON, Canada, in 2007.

He was an Associate Research Scientist and NSERC Post-Doctoral Fellow in information systems at Stern Business School, New York University, New York, NY, USA. He is an Assistant Professor of Computer Science at the University of Central Arkansas, Conway, AR, USA, and the Founding Director of Data Analytics Laboratory. His current research interests include data mining, machine learning, and related applications.

Prof. Sheng is a Lifetime Member of ACM. He was the recipient of the Best Paper Award Runner-Up from KDD'08, and the Best Paper Award from ICDM'11. He is a PC Member for a number of international conferences and a reviewer for several international journals.