# Learning with Unsure Responses

**Kunihiro Takeoka**
NEC Corporation
k_takeoka@nec.com

**Yuyang Dong**
NEC Corporation
dongyuyang@nec.com

**Masafumi Oyamada**
NEC Corporation
oyamada@nec.com

## Abstract

Many annotation systems provide to add an unsure option in the labels, because the annotators have different expertise, and they may not have enough confidence to choose a label for some assigned instances. However, all the existing approaches only learn the labels with a clear class name and ignore the unsure responses. Due to the unsure response also account for a proportion of the dataset (e.g., about 10-30% in real datasets), existing approaches lead to high costs such as paying more money or taking more time to collect enough size of labeled data. Therefore, it is a significant issue to make use of these unsure.

In this paper, we make the unsure responses contribute to training classifiers. We found a property that the instances corresponding to the unsure responses always appear close to the decision boundary of classification. We design a loss function called *unsure loss* based on this property. We extend the conventional methods for classification and learning from crowds with this *unsure loss*. Experimental results on real-world and synthetic data demonstrate the performance of our method and its superiority over baseline methods.

## 1 Introduction

Preparing a sufficient amount of "appropriately labeled" datasets is key to build a high-quality supervised machine learning model. So far, such labeling process has relied on expert annotators or wisdom of crowds (Dawid and Skene 1979; Whitehill et al. 2009; Welinder et al. 2010; Jin et al. 2017). However, those annotators are not always being correct and the resulting datasets sometimes contain incorrectly labeled instances in practice.

Conventional approaches try to avoid incorrect labeling by allowing annotators to give up a labeling task if it is too difficult for them (Zhong, Tang, and Zhou 2015; Ding and Zhou 2018). In these approaches, annotators are permitted to choose the third option called "unsure" instead of ordinary "Yes" or "No" when they are not so confident for the label of the instance. Thus, conventional approaches can just discard the instances with "unsure", and train a high-quality machine learning model with the remained high-confident labeled instances.

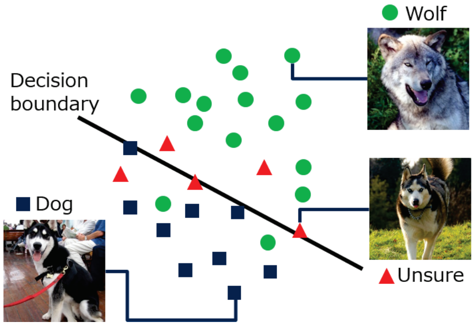

Figure 1: In our dog-wolf classification task, instances corresponding to unsure responses from annotators (red triangles) are located close to the ground-truth decision boundary.

In reality, however, a fair amount of instances are labeled as "unsure". According to the real dataset (UDI Twitter Crawl-Aug2012) in the related work (Zhong, Tang, and Zhou 2015), there are about 32% "unsure" in all responses. Moreover, we also conduct a preliminary experiment on classifying the images of dogs and wolfs, and the "unsure" instances make up 15.6% of all responses. In these cases, with the conventional "let's ignore unsure instances" strategy, the size of the training dataset is decreased, which may cause the issues such as (1) small training sets incur an overfitting problem, and (2) need an extra labeling process that costs much and takes time.

Naturally, this comes up with an important research question: **can we leverage "unsure" instances instead of ignoring them to train a high-quality model?**

In this paper, we examine the behavior of the "unsure" instances in real-world labeled datasets, we found that the "unsure" instances tend to be located close to the ground truth decision boundary (Section 2.1). Figure 1 shows the concept of the relationship between "unsure" instances and the decision boundary. It implies that "unsure" instances can be a strong signal for determining the position of the decision boundary, that is, they can help training a high-quality supervised model. Based on this observation, we first define two learning problems with unsure responses (Section 2.2). Then, we design a novel loss function named *unsure loss* (Section 3.1). The unsure loss works like a regulariza-

Table 1: Main symbols and description.

| Symbol | Description |
|---|---|
| $X$ | set of instances |
| $\mathcal{J}$ | set of annotators |
| $j \in \mathcal{J}$ | annotator $j$ |
| $x_i \in \mathcal{X}$ | instance $i$ |
| $y_i^j$ | annotated label of instance $i$ by annotator $j$ |
| $D_j$ | labeled dataset by annotator $j$ without *unsure* |
| $U_j$ | labeled instances with *unsure* by annotator $j$ |
| $f : \mathcal{X} \to \mathbb{R}$ | score function of personal classifier |
| $\theta_j$ | parameter of the $j$-th personal classifier |
| $\theta_0$ | parameter of the global classifier |
| $\lambda, \gamma, \eta$ | hyper-parameters of our model |
| $L(\cdot)$ | original loss function |
| $H(\cdot)$ | unsure loss |

tion term that makes the estimated decision boundary close to the "unsure" instances. We extend conventional methods with our unsure loss and propose solutions for two classification problems (Sections 3.2 and 3.3). Through our experiments (Section 4), we confirmed that our "unsure loss" approach empirically outperforms the conventional "let's ignore unsure instances" approaches in many datasets.

Our contributions are summarized as follows:

- We find that the instances corresponding to unsure responses by annotators are close to the ground truth decision boundary.
- We propose an *unsure loss* to consider the unsure responses for learning a classifier.
- We extend the conventional methods for the problems of classification and learning from crowds by using the unsure loss to improve the performance.
- Experiments on both synthetic and real-world datasets show the performance of our approach is better than those of the conventional model.

## 2 Unsure Responses

In this section, we first introduce our assumption and the observation of a preliminary experiment with unsure responses. Then, we define the problems of learning with unsure responses, which are tackled in this paper. The main symbols in this paper are summarized in Table 1.

### 2.1 Assumption and Observation

For annotators, it is easy to annotate an instance if it is clear to distinguish the class which this instance belongs to. On the other hand, if the annotator does not have enough knowledge on candidate classes of the instance, annotators may feel hard to add a label. When using the labeled instances to build a classifier with machine learning, those hard-to-label instances may become a low-quality training data if annotators are forced to label them. However, if we allow annotators to annotate them as "unsure", these hard-to-label instances may help us to build a better classifier, since these instances have all the features of the candidate class. Therefore, we assume that the unsure instances located close to the ground truth decision boundary. Figure 1 is an illustrator to help to understand our assumption.

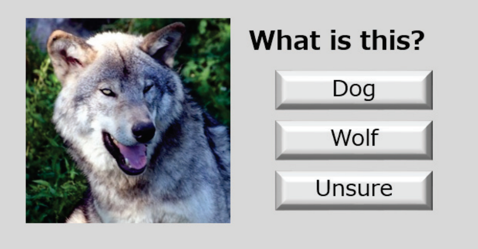

Figure 2: Our annotation system for binary image classification tasks.

Table 2: The number of all responses and the unsure responses with five annotators for the 400 dog-wolf images classification task.

| Annotator | # Responses | # unsure |
|---|---|---|
| #1 | 400 | 45 |
| #2 | 200 | 18 |
| #3 | 400 | 123 |
| #4 | 25 | 0 |
| #5 | 400 | 37 |

To test the above assumption, we conducted a preliminary experiment in our annotation system. We used 400 images of dogs and wolves from an open-source dataset ImageNet (Deng et al. 2009) and asked five annotators to classify them. As shown in Figure 2, we allowed the annotators to choose an unsure option when they feel hard to classify.

We first observed the number of unsure responses for each annotator. As shown in Table 2, we collected 1425 responses in total and there are 223/1425 $\simeq$ 15.6% unsure responses. Then, we also took a deep observation that calculating the distance between all responses and the ground truth decision boundary, then sorted them from near to far, and summarized in Figure 3. In this figure, the dashed line represents that the unsure responses distributed uniformly according to the decision boundary. However, all solid lines of annotators are located on the left up above the dashed line, this means unsure responses tend to distribute around to the decision boundary rather than a uniform distribution. To our surprise, we observed that there are 114 unsure responses in the 160 nearest neighbor instances to the decision boundary. In other words, the majority (114 / 223 $\simeq$ 51.1%) of unsure responses are located in a close range (160 / 400 = 40%) to the ground truth decision boundary.

### 2.2 Problem Definition

When training a classifier with the labeled dataset of annotators' responses, the unsure responses can be used to make a better performance since they are located close to the ground truth decision boundary. We defined two problems on "learning with the unsure responses", one targets to a single annotator, and another one targets to multiple annotators.

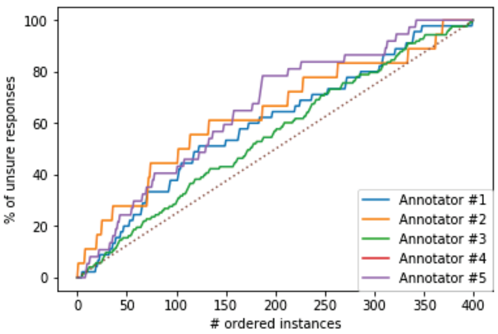

Figure 3: The proportion of the unsure responses (vertical axis) from near to far according to the ground truth decision boundary.

**Learning a Classifier with Unsure Responses** First, we define the problem of learning a classifier with unsure responses by a single annotator. Let $j$ denote an annotator. Let $\mathcal{X} = \{x_1, x_2..., x_N \mid x \in \mathcal{X}\}$ denote $N$ instances to be classified, and $D_j = \{(x_i, y_i^j) \mid y_i^j \in \{-1, 1\}\}_{i=1}^N$ denote the labeled dataset by $j$. Note that $\{-1, 1\}$ mean binary classification labels. $U_j = \{x_1^u, \ldots x_m^u\}$ denote a set of instances with the unsure responses by $j$. In the rest of the paper, "instances with the unsure responses" is called *unsure instances* by abbreviation. Note that the each unsure instance potentially belongs to a class of $\{-1, 1\}$.

We formalize the problem of learning a classifier with unsure responses by a single annotator as follows:

**Problem 1** (Learning a Classifier with Unsure Responses)**.** *Given a set of labeled dataset $D_j$ by an annotator $j$, and a set of unsure instances $U_j$ by an annotator $j$, estimate the classifier $f : \mathcal{X} \to \mathbb{R}$ that accurately predicts the best label of an instance.*

**Learning from Crowds with Unsure Responses** Then, we define the problem of learning from crowds with unsure responses by multiple annotators. Let $I_j \subseteq \{1, \ldots, N\}$ be the index set of labeled instances by annotator $j$, and $I_j^u \subseteq \{1, \ldots, N\}$ be the index set of unsure instances by annotator $j$. Let $D_j = \{(x_i, y_i^j) \mid i \in I_j\}$ denote the pair of instances and annotated labels by the annotator $j$, and $U_j = \{x_i \mid i \in I_j^u\}$ denote the instances corresponding to unsure responses by the annotator $j$.

We formalize the problem of learning from crowds with unsure responses by multiple annotators as follows:

**Problem 2** (Learning from Crowds with Unsure Response)**.** *Given a set of labeled data $\{D_j\}_{j=1}^J$ and instances with unsure responses $\{U_j\}_{j=1}^J$ annotated by annotators $\mathcal{J}$, estimate the global classifier that predicts the best label of an instance.*

# 3 Learning with Unsure Responses

Inspired by the assumption in Section 2.1, we propose a series of methods LEUR-*, which train classifiers with the unsure responses from annotators. We first design a loss function named *unsure loss*. Then, by using the unsure loss, we give two concrete LEUR models to solve Problems 1 and 2.

## 3.1 Unsure Loss

Training a classifier on a set of labeled instances $D$ is modeled as *Empirical Risk Minimization*:

$$\min_{\theta} \sum_{(x_i, y_i) \in D} L(y_i, f(x_i; \theta)), \tag{1}$$

where $L$ is a loss function and $f$ is a score function that measures the distance between the instance $x$ and the estimated decision boundary $\theta$. Based on our assumption that unsure responses $x^u \in U$ are close to the decision boundary, we design a component in empirical risk minimization called *unsure loss* in the form of

$$H_{sq}(U, \theta) = \sum_{x^u \in U} f(x^u; \theta)^2. \tag{2}$$

The unsure loss measures distances between the unsure instances $U$ and the estimated decision boundary, and regularize the decision boundary to be placed close to the unsure instances.

Besides the square function, unsure loss can take a variety of forms such as

$$\begin{aligned} & H_{log}(U, \theta) \\ & = -\sum_{x^u \in U} (\log \sigma(f(x^u; \theta)) + \log(1 - \sigma(f(x^u; \theta)))) \end{aligned} \tag{3}$$

and

$$H_{abs}(U, \theta) = \sum_{x^u \in U} |f(x^u; \theta)|. \tag{4}$$

In the experiments section, we examine the performance difference of unsure losses.

## 3.2 Learning a Classifier with Unsure Response

To solve the Problem 1, we propose a method for learning a classifier with the unsure responses. In this method, the unsure loss is utilized to improve performance.

Problem 1 can be formalized as that optimizing the Equation (5) with the parameter $\theta$

$$\min_{\theta_j} \sum_{(x_i, y_i^j) \in D_j} L(y_i^j, f(x_i; \theta_j)) + \gamma H(U_j, \theta_j). \tag{5}$$

The unsure loss is similar to a regularization term to avoid overfitting.

For example, we can use the logistic regression model to solve Problem 1 with the unsure loss. The original loss function in Equation (6) is called log-loss. The unsure loss $H(.)$ calculates the loss between the decision boundary and the projected instances.

$$L(y, v) = -y \log \sigma(v) - (1 - y) \log(1 - \sigma(v)) \tag{6}$$

Training a classifier with the unsure loss is model-agnostic because the unsure loss does not depend on a classification model. For instance, when using a logistic regression model as the classifier, the training loss $L(y, \hat{y})$ is chosen by the score function $f$ so that we can only take log-loss as a training loss function $L$. But for the unsure loss, we can take an appropriate one based on the feature of the unsure responses in the dataset. The classifier is trained with both labeled instances and unsure instances with the help of unsure loss. Therefore, the unsure loss can also help to avoid underfitting and overfitting if the labeled instances have a small number.

### 3.3 Learning from Crowds with Unsure Response

LEUR can also be used in the problem of learning from crowds. LEUR can extend the conventional methods and improve the predictive performance by adding the unsure loss when learning the personal classifiers corresponding to annotators.

We propose two methods of extension for learning from crowds with unsure responses: LEUR-naïve and LEUR-PC. LEUR-naïve is a simple approach that using LEUR method for learning from crowds straightforwardly. LEUR-PC extends the personal classifier model in (Kajino, Tsuboi, and Kashima 2012) with LEUR method.

**LEUR-naïve** For the LEUR-naïve method, we formalize the Problem 2 as the following Equation (7). According to the Equation (7), we can minimize each loss and find the optimal parameter of the personal classifiers $\Theta = \{\theta_j\}$, by using not only labeled dataset $D_j$ but also the instances corresponding to unsure responses $U_j$. We add a new loss function $H(U_j, \theta_j)$ for utilizing the unsure responses.

$$\min_{\Theta} \sum_{j\in\mathcal{J}} L(D_j,\theta_j) + \gamma H(U_j,\theta_j) + \lambda R(\theta_j) \qquad (7)$$

Each loss function of an annotator consists of three components: training loss $L(D_j, \theta_j)$, regularization term $R(\theta_j)$ and unsure loss $H(U_j, \theta_j)$. Let use a SVM (support verctor machine) as an example, we can take SVM as the personal classifiers and extend the loss of SVM by using the unsure loss. The original loss of SVM $L(D_j, \theta_j)$ is in Equation (8). The unsure loss of SVM $H(U_j, \theta_j)$ is Equation (2). Besides SVM, we can also use unsure loss in other classifiers.

$$L(D_j,\theta_j) = \sum_{(x_i,y_i^j)\in D_j} \max(0, 1 - y_i^j \cdot f(x_i;\theta_j)) \qquad (8)$$

$$R(\theta_j) = ||\theta_j||^2 \qquad (9)$$

**LEUR-PC** LEUR-PC is an extension of the PC (personal classifier) method in (Kajino, Tsuboi, and Kashima 2012) with our unsure loss. The PC method trains a personal classifier for the responses from each annotator, and use these classifiers to build a global classifier model for classification. We formalize the approach by using the following Equation (10). $L$ is the loss function of a personal classifier $\theta_j$ for labeled dataset $D_j$. $H$ is the unsure loss function for the unsure responses $U_j$. The original PC method has a regularization term $R$ for the parameters of global classifier $\theta_0$ and personal classifiers $\theta_j$.

$$\min_{\Theta} \sum_{j\in\mathcal{J}} L(D_j,\theta_j) + \eta R(\theta_j,\theta_0) + \gamma H(U_j,\theta_j) \qquad (10)$$

Since the original PC model estimates the parameters of all personal classifier then builds a global classifier, it needs sufficient numbers of labeled responses from each annotator. The proposed LEUR-PC extends the PC model with our unsure loss so that the unsure responses can also be used to train personal classifiers. Therefore, when the labeled responses are insufficient, LEUR-PC can still use the unsure responses to infer a good decision boundary for a global classifier.

**Estimating Importance of Annotators with Unsure Responses** We also assume that the unsure responses also react to the importance of the annotators. As annotators may have different numbers of their unsure responses, it is natural to think that an annotator with less unsure responses is an expert. On the other hand, for the same instance set, more unsure responses lead to less performance of learning a personal classifier. Therefore, we consider the number of unsure responses as the weight of the importance of an annotator. We can use these weighted personal classifiers to improve the performance of the global classifier as follows:

$$\min_{\Theta} \sum_{j\in\mathcal{J}} L(D_j,\theta_j) + \eta_j R(\theta_j,\theta_0) + \gamma H(U_j,\theta_j) \qquad (11)$$

where $\eta_j$ is a hyper-parameter to adjust the number of unsure responses. For example, we can set the the hyper-parameter $\eta_j = \frac{|D_j|+|U_j|}{|U_j|+1}$.

### 3.4 Extension to Complicated Problems

Besides binary classification, it is worthy to note that our unsure loss can also apply in multi-label and multi-class classification problems. Specifically, we can transfer a multi-label problem to multiple binary classification problems by binary relevance, and the unsure loss can be applied to these binary classification problems. We can also transform a multi-class problem to binary class problems by one-vs.-one reduction.

## 4 Experiments

To evaluate the effectiveness of our proposed methods, we conduct experiments on both a synthetic dataset and a real dataset. We answer the following questions in our experiments:

- Can unsure responses help to increase the predictive performance of classifiers by using our proposed unsure loss?
- In what kind of situation that the unsure loss has a strong effect on predictive performance?

### 4.1 Datasets and Setting

**Synthetically Labeled Data** For synthetically labeled data, we picked up 700 images as training data and 7000 images as testing data, which representing “0” or “6” from the MNIST (LeCun et al. 1998) image dataset. The features are extracted by PCA that converting data into a lower-dimensional space. The synthetic labels are generated based on the ground truth labels by a labeling process. We used two different labeling process: DS labeling process (Dawid and Skene 1979) and PC labeling process (Kajino, Tsuboi, and Kashima 2012). These processes are used to simulate the behavior of noisy annotators (Raykar et al. 2010; Kajino, Tsuboi, and Kashima 2012; 2013; Atarashi, Oyama, and Kurihara 2018; Zhang, Wu, and Sheng 2019). We adjust these two processes and make them can generate unsure responses.

- **DS labeling process.** Every synthetic annotator has two parameters $\alpha \in [0, 1]$ and $\beta \in [0, 1]$ that represent the expertise to either of the class. A class label to an instance is generated by flipping a coin with the probability that equals to the expertise of the annotators. We used the most common parameters: $\alpha = \beta = 0.8$. On the other hand, an unsure label to an instance is generated by according their distance and a parameter $\rho$ to the decision boundary. We take the top-$\rho$ instances and label them as unsure.
- **PC labeling process.** Different from the DS labeling process that all synthetic annotators share a common decision boundary, PC labeling process creates classifiers with different decision boundaries for all synthetic annotators. Labels are generated based on the annotators' classifiers. For each annotator, we take top-$\rho$ instances that are close to the decision boundary and label them as unsure.

**Human Labeled Data** We collected a real dataset named "DOG-WOLF" with 1500 responses from five human annotators. We picked up 200 dog images and 200 wolf images from the ImageNet Dataset (Deng et al. 2009) and asked annotators to determine whether an image is a dog or a wolf. Annotators can select a label for the image from "dog", "wolf" and "unsure" as shown in Figure 2. We used pretrained AlexNet to extract the features. In the annotation, the five annotators have 98%, 96%, 88%, 86%, 60% accuracies, and have 92% of average accuracy, with rho = 0.156. The inter-annotator agreement value is 0.7453 by the Fleiss' kappa method (Fleiss 1971), and it indicates a good agreement.

To evaluate the effectiveness of *unsure loss*, we conducted the following two tasks in our experiments.

**Learning from Noisy Labels with Unsure Responses** Given a collection of noisy and unsure instances of synthetic labels, we trained the following binary classifiers and compared the performance:

- **LR**: Logistic regression that discards the unsure responses.
- **SVM**: Support vector machine that discards the unsure responses.
- **LEUR**: The proposed method in Section 3.2, which uses unsure responses as a regularizer.

**Learning from Crowds with Unsure Responses** Given a collection of noisy and unsure instances generated from a crowd of above synthetic annotators, we trained the following binary classifiers and compared the performance:

- **Raykar**: A popular learning from crowds method, which applies DS model to determine the classifier (Raykar et al. 2010).
- **PC**: A state-of-the-art learning from crowds method (Kajino, Tsuboi, and Kashima 2012) where each worker (annotator) is modeled as a classifier.
- **SSPC**: A semi-supervised personal classifier. To make it comparable, we made an extension to PC so that it considers unsure responses in its training in a semi-supervised manner.
- **LEUR-PC**: The proposed method in Section 3.3, which uses unsure responses with our unsure loss.

Table 3: Comparison results of baselines and our proposed methods on synthetic data with a single synthetic annotator (unsure ratio $\rho = 20\%$).

| Method | F1 | AUC |
|---|---|---|
| LR | 0.85±0.03 | 0.76±0.03 |
| LEUR-LR-Log | 0.87±0.02 | 0.79±0.02 |
| LEUR-LR-Abs | **0.88±0.01** | **0.80±0.01** |
| LEUR-LR-Sq | 0.84±0.04 | 0.79±0.01 |
| SVM | 0.85±0.06 | 0.75±0.05 |
| LEUR-SVM-Log | 0.84±0.07 | 0.75±0.05 |
| LEUR-SVM-Abs | **0.87±0.01** | **0.79±0.02** |
| LEUR-SVM-Sq | 0.86±0.01 | 0.76±0.03 |

**Detailed Setting** The hyper-parameters are tuned with the validation data. We used Adam (Adaptive moment estimation) optimization algorithm in our experiment. Note that LEUR-LR-Sq means a method of logistic regression model that extended by our squared unsure loss with Equation (2). In all experiments, the performance of the methods is measured by AUC (Area Under the Curve) and F1-score which are known as appropriate measures in the binary classification problems.

## 4.2 Experimental Results on Synthetic Data

**Learning from Noisy and Unsure Responses** We first tested the effectiveness of the unsure loss on a synthetic dataset with a single annotator. In this experiment, the synthetic dataset is generated by the DS model from the MNIST dataset with $\rho = 0.2$ rate of unsure labels. We used 100 instances as the training data, and test the performance with other data.

According the results in Table 3, the unsure loss did help the classifier to increase the predictive performance. We found that the Square unsure loss works best, this is because Square is neither too sensitive nor too insensitive for the unsure responses compared to Log and Abs. We also found that the hyper-parameter $\gamma$ is also important. if we set $\gamma$ too large (i.e., 100), original loss function will be ignored and the classifier becomes hard to train. On the contrary, if we set $\gamma$ too small (i.e., $1e^{-7}$), the unsure loss will be ignored and there is no benefit from the unsure loss (i.e., the same performance to the original model). Therefore, we found that the performance seems appropriate to set the $\gamma$ in the value range of $\frac{1}{|U|} < \gamma \leq 1$.

**Learning from Crowds with Unsure Responses (DS).** We also tested the effectiveness of the unsure loss on multiple annotators of learning from crowds. We set the number

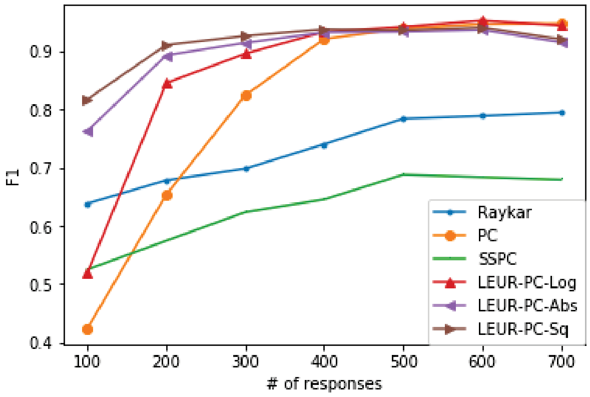

Figure 4: Comparison results on MNIST data with 10 synthetic DS annotators, varying total results number with $\rho = 20\%$ unsure responses from annotators.

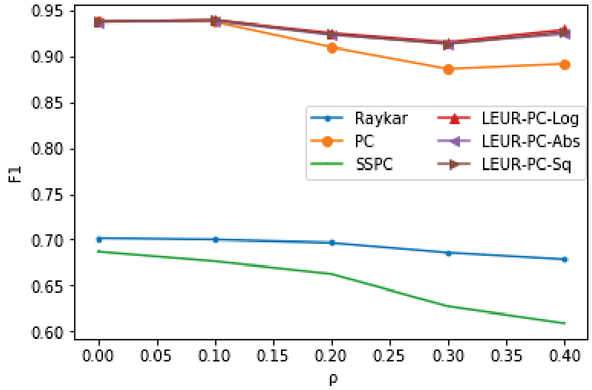

Figure 5: Comparison results on MNIST data with 10 synthetic DS annotators, varying unsure responses number with 500 total responses.

Table 4: Comparison results on MNIST data with 10 synthetic DS and PC annotators. We set the number of responses from each annotator as 30, and the number of unsure responses as 12 (20% of responses).

| | DS | | PC | |
|---|---|---|---|---|
| Method | F1 | AUC | F1 | AUC |
| Raykar | 0.69 ± 0.03 | 0.78 ± 0.04 | 0.77 ± 0.13 | 0.85 ±0.07 |
| PC | 0.82 ± 0.10 | 0.98 ± 0.01 | 0.78 ± 0.20 | 0.94 ±0.14 |
| SSPC | 0.63 ± 0.07 | 0.72 ± 0.08 | 0.61 ± 0.12 | 0.65 ±0.12 |
| LEUR-PC-Log | 0.90 ± 0.06 | **0.99 ± 0.01** | **0.83 ± 0.13** | **0.96±0.07** |
| LEUR-PC-Abs | 0.92 ± 0.03 | 0.97 ± 0.02 | 0.82 ± 0.11 | 0.94 ± 0.04 |
| LEUR-PC-Sq | **0.93 ± 0.03** | 0.98 ± 0.02 | 0.82 ± 0.03 | 0.95 ± 0.06 |

of annotator as 10, and generate the synthetic dataset with DS annotators from MNIST dataset with $\rho = 0.2$ rate of unsure labels.

Figure 4 shows the comparison result of predictive performance with Raykar, PC, SSPC and proposed LEUR-PC methods. Note that the horizontal axis is the total number of responses from 10 annotators. The proposed LEUR-PC methods extremely outperformed than baseline methods when the responses set is small (100 - 300). This is strong proof of our assumption and the effectiveness of our proposed methods. With the response number increasing, the PC method catches up with our method with similar performance. This is because the number of high-quality labels also increased and make PC performed better. However, SSPC degrades its performance when the number of responses exceeds 500. SSPC may treat more unsure labels as unlabeled instances, then the learned decision boundary biases to either of the class, leading to incorrect results. Back to the proposed method with another view, a smaller number of responses make less cost and less time on collecting training data, which is also a strong point of our proposed methods.

In Figure 5, we observed the effect of the different number of unsure responses. The total response is a fixed number of 500 and the $\rho$ is from 0 to 0.4. We found that our proposed methods keep superiority against the other baselines. According to Figures 4 and 5, we found that both the total sample size and rho are crucial factors, this is because they can affect the effective sample size. When the effective sample size is insufficient, our proposed method with unsure loss can have an extremely better performance than baselines.

**Learning from Crowds with Unsure Responses (DS vs PC)** We also compared the F1 score and AUC with the synthetic data with 10 annotators based on DS and PC labeling process. According to the results in Table 4, in DS labeling process, the proposed LEUR-PC methods have at least 0.08 more F1 score and 0.01 more AUC than the baseline methods; and in PC labeling process, the proposed LEUR-PC methods have at least 0.05 more F1 score and 0.02 more AUC than the baseline methods. We found that there is no big difference between DS and PC labeling process.

### 4.3 Results on Human-Labeled Real Data

**Learning from Noisy and Unsure Responses** For the real dataset DOG-WOLF, we first evaluated the unsure loss. We take 100 images, which is the 25% responses from an annotator, as the training data, and use the remained 300 images as the test data. Table 5 summarizes the comparison results of the proposed LEUR methods with baseline LR and SVM methods. These results proved that our proposed unsure loss also works on the real dataset. We also found the proposed

Table 5: Comparison results on the real DOG-WOLF dataset annotated by an single human annotator.

| Method | F1 | AUC |
|---|---|---|
| LR | 0.80±0.05 | 0.81±0.01 |
| LEUR-LR-Log | 0.84±0.04 | 0.81±0.01 |
| LEUR-LR-Abs | **0.85±0.04** | **0.84±0.04** |
| LEUR-LR-Sq | 0.84±0.04 | **0.84±0.04** |
| SVM | 0.74±0.06 | 0.74 ±0.05 |
| LEUR-SVM-Log | 0.74 ±0.06 | 0.74 ±0.05 |
| LEUR-SVM-Abs | **0.78 ± 0.06** | **0.78 ±0.05** |
| LEUR-SVM-Sq | 0.76 ±0.06 | 0.76 ±0.06 |

Table 6: Comparison results on the real DOG-WOLF dataset annotated by five human annotators.

| Method | F1 | AUC |
|---|---|---|
| Raykar | 0.51±0.07 | 0.49±0.06 |
| PC | 0.60±0.08 | 0.50±0.07 |
| SSPC | 0.51±0.03 | **0.53±0.03** |
| LEUR-PC-Log | **0.63±0.06** | 0.51±0.06 |
| LEUR-PC-Abs | 0.62±0.05 | 0.51±0.06 |
| LEUR-PC-Sq | **0.63±0.07** | 0.52±0.03 |

methods with the absolute unsure loss have the best performance overall methods on both F1 score and AUC.

**Learning from Crowds with Unsure Responses** Secondly, we evaluated the unsure loss on the real DOG-WOLF dataset in the learning from crowds setting. We took 100 images, which is the 25% responses from each annotator, as the training data for the personal classifier, and use the remained 300 images as the test data. Table 6 summarizes the comparison results of the proposed LEUR-PC methods with baseline Raykar, PC, and SSPC methods. The proposed LEUR-PC methods outperformed other methods in F1 scores and competitive in AUC scores. These results proved that our proposed unsure loss also works well in learning from crowd settings on the real data.

## 5 Related Work

**Unsure Option in Annotation** (Zhong, Tang, and Zhou 2015) proposed a method for filtering the instances which have a high probability to be labeled as "unsure" by annotators from the unlabeled dataset. Specifically, the method estimates the parameters of a classifier and reliability models simultaneously along with the active learning procedure. The reliability model is updated with the current unsure responses to avoid getting the unsure responses in the future procedure. Different from our work, the method in (Zhong, Tang, and Zhou 2015) tried to avoid the unsure responses. This is because the unsure responses are received incrementally in the active learning procedure, and it cannot help to imply the decision boundary more accurately. While in our work, we can get all unsure responses in advance and make use of them to imply the decision boundary.

(Ding and Zhou 2018) focused on the problem of saving cost by selecting high-quality labeled instances with a threshold value calculated by a confidence model. The unsure responses are used to build the confidence model to select high-quality label instances, while in our work, we used the unsure responses as special labeled instances that imply the decision boundary.

**Learning from Crowds** The works of learning from crowds are about training a classifier model by using the labeled dataset collected from crowdsourcing services (Zhang, Wu, and Sheng 2016). (Raykar et al. 2010) formalized the problem of learning from crowds, and proposed a solution based on the DS model (Dawid and Skene 1979) for workers. They combined the logistic regression model and the DS model to estimate the best parameters of the classifier model from the features of instances and the noisy labeled responses of annotators.

(Kajino, Tsuboi, and Kashima 2012; 2013) proposed a PC (Personal Classifier) model. They assumed that each annotator is modeled by a personal classifier and learned optimal parameters with the responses from annotators.

(Atarashi, Oyama, and Kurihara 2018) proposed a semi-supervised method for learning from crowds. The method utilizes the unlabeled data to improve predictive performance. They assumed that the unlabeled data are sampled from the original distribution of the data. However, the unsure responses in our problem are not followed the data distribution.

**Universum** Universum is a set of classes which are not originally considered in the classification. For example, in a dog-wolf classification, "cat" can be a universum. (Weston et al. 2006) found that those univesum sets sometimes help to determine the ground truth decision boundary of the original classes, and proposed a regularization with the universum set. Our problem of unsure responses, which consists of hard-to-distinguish instances (e.g., wolf-looking dogs and dog-looking wolves), is different from the problem of universum set. Even the universum regularization has a similar form with our *unsure loss*, it can not deal with the personal classifiers which consider the number of unsure responses as a weight of importance (Section 3.3).

(Sinz et al. 2007) implied that the universum regularization performs well if the universum instances are positioned "in-between" of the original classes. Since we assumed the unsure responses to be placed ‘in-between‘ the decision boundary in Section 2.1, this analysis supports why our proposed unsure loss works well empirically.

**Quality Control in Crowd-sourcing** Quality control in crowdsourcing (QCC) is a research field for inferring the ground truth labels from the annotators' response. (Welinder et al. 2010) proposed a method for QCC which focuses on modeling the interaction between the difficulty of tasks and the ability of annotators. (Jin et al. 2017) proposed a method utilizing side information of instances and annotators for the QCC problem. (Oyama et al. 2013) proposed a QCC method that considers the confidence scores corresponding to the annotators' responses. (Dumitrache, Aroyo, and Welty 2019) studied on recognizing different levels of ambiguity in the labeled datasets.

## 6 Conclusion

We studied the problem of learning a classifier with the unsure responses. We designed an unsure loss function to use these unsure responses to imply the decision boundary. We extended the conventional methods for classification with the unsure loss to improve the predictive performance. Extensive experiments on real and synthetic datasets showed the superiority of our proposed methods over the baselines.

## References

Atarashi, K.; Oyama, S.; and Kurihara, M. 2018. Semi-supervised learning from crowds using deep generative models. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, AAAI-18*, 1555–1562.

Dawid, A. P., and Skene, A. M. 1979. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 28(1):20–28.

Deng, J.; Dong, W.; Socher, R.; Li, L.; Li, K.; and Li, F. 2009. Imagenet: A large-scale hierarchical image database. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR-09*, 248–255.

Ding, Y., and Zhou, Z. 2018. Crowdsourcing with unsure option. *Machine Learning* 107(4):749–766.

Dumitrache, A.; Aroyo, L.; and Welty, C. 2019. A crowdsourced frame disambiguation corpus with ambiguity. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019*, 2164–2170.

Fleiss, J. L. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin* 76(5):378.

Jin, Y.; Carman, M. J.; Kim, D.; and Xie, L. 2017. Leveraging side information to improve label quality control in crowd-sourcing. In *Proceedings of the Fifth AAAI Conference on Human Computation and Crowdsourcing, HCOMP-17*, 79–88.

Kajino, H.; Tsuboi, Y.; and Kashima, H. 2012. A convex formulation for learning from crowds. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, AAAI-12*.

Kajino, H.; Tsuboi, Y.; and Kashima, H. 2013. Clustering crowds. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence, AAAI-13*.

LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P.; et al. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* 86(11):2278–2324.

Oyama, S.; Baba, Y.; Sakurai, Y.; and Kashima, H. 2013. Accurate integration of crowdsourced labels using workers' self-reported confidence scores. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence, IJCAI-13*, 2554–2560.

Raykar, V. C.; Yu, S.; Zhao, L. H.; Valadez, G. H.; Florin, C.; Bogoni, L.; and Moy, L. 2010. Learning from crowds. *J. Mach. Learn. Res.* 11:1297–1322.

Sinz, F. H.; Chapelle, O.; Agarwal, A.; and Schölkopf, B. 2007. An analysis of inference with the universum. In *Advances in Neural Information Processing Systems, NeurIPS-07*, 1369–1376.

Welinder, P.; Branson, S.; Belongie, S. J.; and Perona, P. 2010. The multidimensional wisdom of crowds. In *Advances in Neural Information Processing Systems 23, NeurIPS-10.*, 2424–2432.

Weston, J.; Collobert, R.; Sinz, F. H.; Bottou, L.; and Vapnik, V. 2006. Inference with the universum. In *Proceedings of the Twenty-Third International Conference on Machine Learning, ICML-06*, 1009–1016.

Whitehill, J.; Ruvolo, P.; Wu, T.; Bergsma, J.; and Movellan, J. R. 2009. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. In *Advances in Neural Information Processing Systems 22, NeurIPS-09.*, 2035–2043.

Zhang, J.; Wu, X.; and Sheng, V. S. 2016. Learning from crowdsourced labeled data: a survey. *Artif. Intell. Rev.* 46(4):543–576.

Zhang, J.; Wu, M.; and Sheng, V. S. 2019. Ensemble learning from crowds. *IEEE Trans. Knowl. Data Eng.* 31(8):1506–1519.

Zhong, J.; Tang, K.; and Zhou, Z. 2015. Active learning from crowds with unsure option. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-15*, 1061–1068.