Consistent Estimators for Learning to Defer to an Expert

Hussein Mozannar [1] **David Sontag** [1]

Abstract

Learning algorithms are often used in conjunction with expert decision makers in practical scenarios, however this fact is largely ignored when designing these algorithms. In this paper we explore how to learn predictors that can either predict or choose to defer the decision to a downstream expert. Given only samples of the expert's decisions, we give a procedure based on learning a classifier and a rejector and analyze it theoretically. Our approach is based on a novel reduction to cost sensitive learning where we give a consistent surrogate loss for cost sensitive learning that generalizes the cross entropy loss. We show the effectiveness of our approach on a variety of experimental tasks.

1. Introduction

Machine learning systems are now being deployed in settings to complement human decision makers such as in healthcare (Hamid et al., 2017; Raghu et al., 2019a), risk assessment (Green & Chen, 2019a) and content moderation (Link et al., 2016). These models are either used as a tool to help the downstream human decision maker: judges relying on algorithmic risk assessment tools (Green & Chen, 2019b) and risk scores being used in the ICU (Futoma et al., 2017), or instead these learning models are solely used to make the final prediction on a selected subset of examples (Madras et al., 2018; Raghu et al., 2019a). A current application of the latter setting is Facebook's and other online platforms content moderation approach (Vincent, 2019; Jhaver et al., 2019): an algorithm is used to filter easily detectible inappropriate content and the rest of the examples are screened by a team of human moderators. Another motivating application arises in health care settings, for example deep neural networks can outperform radiologists in detecting pneumonia from chest X-rays (Irvin et al., 2019), however, many obstacles are limiting complete automation, an intermediate step to automating this task will be the use of models as triage tools to complement radiologist expertise. Our focus in this work is to give theoretically sound approaches for machine learning models that can either predict or defer the decision to a downstream expert to complement and augment their capabilities.

The learned model should adapt to the underlying human expert in order to achieve better performance than deploying the model or expert individually. In situations where we have limited data or model capacity, the gains from allowing the model to focus on regions where the expert is less accurate are expected to be more significant. However, even when data or model capacity are not concerns, the expert may have access to side-information unavailable to the learner due to privacy concerns for example, the hard task is then to identify when we should defer without having access to this side-information. We will only assume in this work that we are allowed access to samples of the experts decisions or to costs of deferring, we believe that this is a reasonable assumption that can be achieved in practical settings. Inspired by the literature on rejection learning (Cortes et al., 2016b), our approach will be to learn two functions: a classifier that can predict the target and a rejector which decides whether the classifier or the expert should predict.

We start by formulating a natural loss function for the combined machine-expert system in section 3 and show a reduction from the expert deferral setting to cost sensitive learning. With this reduction in hand, we are able to give a novel convex surrogate loss that upper bounds our system loss and that is furthermore consistent in section 4. This surrogate loss settles the open problem posed by Ni et al. (2019) for finding a consistent loss for multiclass rejection learning. Our proposed surrogate loss and approach requires only adding an additional output layer to existing model architectures and changing the loss function, hence it necessitates minimal to no added computational costs. In section 5, we show the limitations of approaches in the literature from a consistency point-of-view and then provide generalization bounds for minimizing the empirical loss. To show the efficacy of our approach, we give experimental evidence on image classification datasets CIFAR-10 and CIFAR-100 using synthetic and human experts based on `CIFAR10H` (Peterson et al., 2019), on a hate speech and

[1] CSAIL, Massachusetts Institute of Technology, Cambridge, MA, USA. Correspondence to: Hussein Mozannar <mozannar@mit.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

offensive language detection task (Davidson et al., 2017), and on classification of chest X-rays with synthetic experts in section 6. To summarize, the contributions of this paper are the following:

- We formalize the expert deferral setup and analyze it theoretically giving a generalization bound for solving the empirical problem.
- We propose a novel convex consistent surrogate loss L_{CE} (6) for expert deferral easily integrated into current learning pipelines.
- We provide a detailed experimental evaluation of our method and baselines from the literature on image and text classification tasks.

2. Related Work

Learning with a reject option, *rejection learning*, has long been studied starting with Chow (1970) who investigated the trade-off between accuracy and the rejection rate. The framework of rejection learning assumes a constant cost c of deferring and hence the problem becomes to predict only if one is $1-c$ confident. Numerous works have proposed surrogate losses and uncertainty estimation methods to solve the problem (Bartlett & Wegkamp, 2008; Ramaswamy et al., 2018; Ni et al., 2019; Jiang et al., 2018). Cortes et al. (2016b;a) proposed a different approach by learning two functions: a classifier and a rejection function and analyzed the approach giving a kernel based algorithm in the binary setting. Ni et al. (2019) tried to extend their approach to the multiclass setting but failed to give a consistent surrogate loss and hence resorted to confidence based methods.

Recent work has started to explore models that defer to downstream experts, Madras et al. (2018) considers an identical framework to the one considered here however their approach does not allow the model to adapt to the underlying expert and the loss used is not consistent and requires an uncertainty estimate of the expert decisions. On the other hand, De et al. (2019) gives an approximate procedure to learn a linear model that picks a subset of the training data on which to defer and uses a nearest neighbor algorithm to defer on new examples, the approach used is only feasible for small dataset sizes and does not generalize beyond ridge regression. Raghu et al. (2019a) considers binary classification with expert deferral, their approach is to learn a classifier ignoring the expert and obtain uncertainty estimates for both the expert and classifier and then defer based on which is higher, we detail the limitations of this approach in section 5. Concurrent work (Wilder et al., 2020) learns a model with the mixtures of expert loss first introduced in (Madras et al., 2018) and defers based on estimated model and expert confidence as in Raghu et al. (2019a). Work on AI-assisted decision making has focused on the reverse setting considered here: the expert chooses to accept or reject the decision of the classifier instead of a learned rejector (Bansal et al., 2019; 2020). Additionally, the fairness in machine learning community has started to consider the fairness impact of having downstream decision makers (Madras et al., 2018; Canetti et al., 2019; Green & Chen, 2019a; Dwork & Ilvento, 2018) but in slightly different frameworks than the ones considered here and work has started to consider deferring in reinforcement learning (Meresht et al., 2020).

A related framework to our setting is selective classification (El-Yaniv & Wiener, 2010) where instead of setting a cost for rejecting to predict one sets a constraint on the probability of rejection; here is no assumed downstream expert. Approaches range from deferring based on confidence scores (Geifman & El-Yaniv, 2017), learning a deep network with two heads, one for predicting and the other for deferring (Geifman & El-Yaniv, 2019) and learning with portfolio theory inspired loss functions (Ziyin et al., 2019). Finally, our work bears resemblance to active learning with weak (the expert) and strong labelers (the ground truth) (Zhang & Chaudhuri, 2015).

3. Problem Formulation

We are interested in predicting a target $Y \in \mathcal{Y} = \{1, \cdots, K\}$ based on covariates $X \in \mathcal{X}$ where $X, Y \sim \mathbf{P}$. We assume that we have query access to an expert M that has access to a domain $\mathcal{Z}$ that may contain additional information than $\mathcal{X}$ to classify instances according to the target $\mathcal{Y}$. Querying the expert implies deferring the decision which incurs a cost $l_{exp}(x, y, m)$ that depends on the target y, covariate x and the expert's prediction m. On the other hand, predicting without querying the expert implies that a classifier makes the final decision and incurs a cost $l(x, y, \hat{y})$ where $\hat{y}$ is the prediction of the classifier. Our goal is to build a predictor $\hat{Y} : \mathcal{X} \to \mathcal{Y} \cup \{\perp\}$ that can either predict or defer the decision to the expert denoted by $\perp$. Our strategy for learning the predictor $\hat{Y}$ will be to learn two separate functions $h : \mathcal{X} \to \mathcal{Y}$ (classifier) and $r : \mathcal{X} \to \{0, 1\}$ (rejector). We can now formulate a natural system loss function L for the system consisting of the classifier in conjunction with the expert:

$$L(h,r) = \mathbb{E}_{(x,y)\sim \mathbf{P}, m \sim M|(x,y)} \, [\, \underbrace{l(x,y,h(x))}_{\text{classifier cost}} \underbrace{\mathbb{I}_{r(X)=0}}_{\text{predict}} \tag{1}$$

$$+ \overbrace{l_{\text{exp}}(x,y,m)}^{\text{expert cost}} \overbrace{\mathbb{I}_{r(x)=1}}^{\text{defer}} \,]$$

The above formulation is a generalization of the learning with rejection framework studied by Cortes et al. (2016b) as by setting $l_{\text{exp}}(x,y,m) = c$ for a constant $c > 0$ the two objectives coincide. In Madras et al. (2018), the loss proposed assumes that the classifier and expert costs are the logistic loss between the target and their predictions in the binary target setting.

While our treatment extends to general forms of expert and classifier costs, we will pay particular attention in our theoretical analysis when the costs are the misclassification error with the target. Formally, we define a $0{-}1$ loss version of our system loss:

$$\begin{aligned} &L_{0-1}(h,r) = \\ &\mathbb{E}_{(x,y)\sim\mathbf{P},m\sim M|(x,y)} \left[\, \mathbb{I}_{h(x)\neq y}\mathbb{I}_{r(x)=0} + \mathbb{I}_{m\neq y}\mathbb{I}_{r(x)=1} \, \right] \end{aligned} \tag{2}$$

One may also assume a constant additive cost function $c(x)$ for querying the expert depending on the instance x making $l_{\text{exp}}(x,y,m) = \mathbb{I}_{m\neq y} + c(x)$; such additive costs can be easily integrated into our analysis.

Our approach will be to cast this problem as a *cost sensitive learning* problem over an augmented label space that includes the action of deferral. Let the random costs $\mathbf{c} \in \mathbb{R}_+^{K+1}$ where for $i \in [K]$, $c(i)$ is the i'th component of $\mathbf{c}$ represents the cost of predicting $i \in \mathcal{Y}$ while $c[K+1]$ represents the cost of deferring to the expert. The goal of this setup is to learn a predictor $h : \mathcal{X} \to [K+1]$ minimizing the cost sensitive loss $\widetilde{L}(h) := \mathbb{E}[c(h(x))]$. For example, giving an instance (x,y,m), our loss (1) is obtained by setting $c(i) = l(x,y,i)$ for $i \in [K]$ and $c(K+1) = l_{\text{exp}}(x,y,m)$.

For the majority of this paper we assume access to samples $S = \{(x_i,y_i,m_i)\}_{i=1}^n$ where $\{(x_i,y_i)\}_{i=1}^n$ are drawn i.i.d. from the unknown distribution $\mathbf{P}$ and m_i is drawn from the distribution of the random variable $M|(X=x_i, Y=y_i)$ and access to the realizations of l_{exp} and l when required .

4. Proposed Surrogate Loss

It is clear that the system loss function (1) is not only nonconvex but also computationally hard to optimize. The usual approach in machine learning is to formulate upper bounding convex surrogate loss functions and optimize them in hopes of approximating the minimizers of the original loss (Bartlett et al., 2006). Work from rejection learning (Cortes et al., 2016b; Ni et al., 2019) suggested learning two separate functions h and r and provided consistent convex surrogate loss functions only for the binary setting. We extend their proposed surrogates for our expert deferral setting for binary labels with slight modifications in appendix C. Consistency is used to prove that a proposed surrogate loss is a good candidate and is often treated as a necessary condition. The issue with the proposed surrogates in (Cortes et al., 2016b) for rejection learning is that when extended to the multiclass setting, it is impossible for them to be consistent as was shown by (Ni et al., 2019). Aside the consistency issue, Ni et al. (2019) found that simple baselines can outperform the proposed losses in practice.

The construction of our proposed surrogate loss for the multiclass expert deferral setting will be motivated via two ways, the first is through a novel reduction to cost sensitive learning and the second is inspired by the Bayes minimizer for the $0{-}1$ system loss (3). Let $g_i : \mathcal{X} \to \mathbb{R}$ for $i \in [K+1]$ and define $h(x) = \arg\max_{i\in[K+1]} g_i$, motivated by the success of the cross entropy loss, our proposed surrogate for cost-sensitive learning $\widetilde{L}_{CE}$ takes the following form:

$$\begin{aligned} &\widetilde{L}_{CE}(g_1,\cdots,g_{K+1},x,c(1),\cdots,c(K+1)) = \\ &-\sum_{i=1}^{K+1} (\max_{j\in[K+1]} c(j) - c(i)) \log\left(\frac{\exp(g_i(x))}{\sum_k \exp(g_k(x))} \right) \end{aligned} \tag{3}$$

The loss $\widetilde{L}_{CE}$ is a surrogate loss for cost sensitive learning that generalizes the cross entropy loss when the costs correspond to multiclass misclassification. The following proposition shows that the loss is consistent, meaning it's minimizer over all measurable functions agrees with the Bayes solution.

Proposition 1. *$\widetilde{L}_{CE}$ is convex in $\mathbf{g}$ and is a consistent loss function for $\widetilde{L}$:*

$$\textit{let } \tilde{\boldsymbol{g}} = \arg\inf_{\mathbf{g}} \mathbb{E}\left[\widetilde{L}_{CE}(\mathbf{g},\mathbf{c})|X=x \right]\textit{, then:}$$
$$\arg\max_{i\in[K+1]} \tilde{\boldsymbol{g}}_i = \arg\min_{i\in[K+1]} \mathbb{E}[c(i)|X=x]$$

Proof of Proposition 1 can be found in Appendix C; $\widetilde{L}_{CE}$ is a simpler consistent alternative to the surrogates derived in (Chen et al., 2019) for cost sensitive learning.

Now we consider when the system loss function is L_{0-1} (3), our approach is to treat deferral as a new class and construct a new label space $\mathcal{Y}^{\perp} = \mathcal{Y} \cup \perp$ and a corresponding distribution $\mathbb{P}(Y^{\perp}|X=x)$ such that minimizing the misclassification loss on this new space will be equivalent to minimizing our system loss L_{0-1}. The Bayes optimal classifier on $\mathcal{Y}^{\perp}$ is clearly $h^{\perp} = \arg\max_{y^{\perp}\in\mathcal{Y}^{\perp}} \mathbb{P}(\mathcal{Y}^{\perp} = y^{\perp}|X=x)$, and we need it to match the decision of the Bayes solution h^B, r^B of L_{0-1} (3):

$$h^B, r^B = \arg\inf_{h,r} L_{0-1}(h,r) \tag{4}$$

where the infimum is over all measurable functions. Denote by $\eta_y(x) = \mathbb{P}(Y=y|X=x)$, it is clear that for $x \in \mathcal{X}$ the best classifier is the same as the Bayes solution for standard classification since if we don't defer we have to do our best. Now we only reject the classifier if it's expected error is higher than the expected error of the expert which we formalize in the below proposition:

Proposition 2. *The minimizers of the loss L_{0-1} (3) are* *defined point-wise for all* $x \in \mathcal{X}$ *as:*

$$
\begin{aligned}
h^B(x) &= \arg\max_{y\in\mathcal{Y}} \eta_y(x) \\
r^B(x) &= \mathbb{I}_{\max_{y\in\mathcal{Y}} \eta_y(x) \le \mathbb{P}(Y=M|X=x)}
\end{aligned}
\tag{5}
$$

Proof of the above proposition can be found in Appendix C and equation (5) give us sufficient conditions for consistency to check our proposed loss. Let $g_y : \mathcal{X} \to \mathbb{R}$ for $y \in \mathcal{Y}$ and define $h(x) = \arg\max_{y\in\mathcal{Y}} g_y$, similarly let $g_\perp : \mathcal{X} \to \mathbb{R}$ and define $r(x) = \mathbb{I}_{\max_{y\in\mathcal{Y}} g_y(x) \le g_\perp}$ the proposed surrogate loss for L_{0-1} (1) in the multiclass setting is then:

$$
L_{CE}(h, r, x, y, m) = -\log\left(\frac{\exp(g_y(x))}{\sum_{y' \in \mathcal{Y}\cup\perp} \exp(g_{y'}(x))}\right) \tag{6}
$$

$$
- \mathbb{I}_{m=y} \log\left(\frac{\exp(g_\perp(x))}{\sum_{y' \in \mathcal{Y}\cup\perp} \exp(g_{y'}(x))}\right)
$$

The proposed surrogate L_{CE} is in fact consistent and upper bounds L_{0-1} as the following theorem demonstrates.

Theorem 1. *The loss* L_{CE} *is convex in* $\mathbf{g}$, *upper bounds* L_{0-1} *and is consistent:* $\inf_{h,r} \mathbb{E}_{x,y,m}[L_{CE}(h, r, x, y, m)]$ *is attained at* (h^*_{CE}, r^*_{CE}) *such that* $h^B(x) = h^*_{CE}(x)$ *and* $r^B(x) = r^*_{CE}(x)$ *for all* $x \in \mathcal{X}$.

Proof of Theorem 1 can be found in Appendix C. When the costs $c(1), \cdots, c(K+1)$ are in accordance with our expert deferral setting the loss $\widetilde{L}_{CE}$ reduces to L_{CE}. Now stepping back and looking more closely at our loss L_{CE}, we can see that the loss on examples where the expert makes a mistake becomes the cross entropy loss with the target. On the other hand, when the expert agrees with the target, the learner faces two opposing decisions whether to defer or predict the target. We can encourage or hinder the action of deferral by modifying the loss with an additional parameter $\alpha \in \mathbb{R}^+$ as $L^{\alpha}_{CE}(h, r, x, y, m)$:

$$
- (\alpha \cdot \mathbb{I}_{m=y} + \mathbb{I}_{m\neq y}) \log\left(\frac{\exp(g_y(x))}{\sum_{y' \in \mathcal{Y}\cup\perp} \exp(g_{y'}(x))}\right) \tag{7}
$$

$$
- \mathbb{I}_{m=y} \log\left(\frac{\exp(g_\perp(x))}{\sum_{y' \in \mathcal{Y}\cup\perp} \exp(g_{y'}(x))}\right)
$$

Note that $L^1_{CE} = L_{CE}$. The effect of α is to re-weight examples where the expert is correct to discourage the learner of fitting them and instead focus on examples where the expert makes a mistake. In practice, one would treat α as an additional hyperparameter to optimize for.

5. Theoretical analysis

In this section we focus on the zero-one system loss function L_{0-1} and try to understand previous proposed solutions in the literature in comparison with our method from a theoretical perspective.

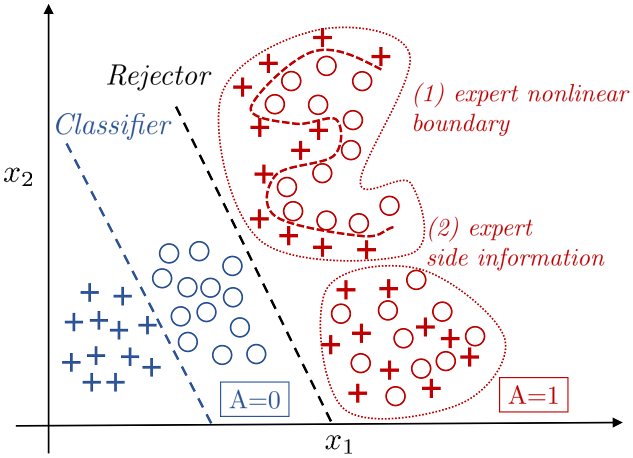


Figure 1. Setting of two groups, red and blue, the task is binary classification with labels $\{o, +\}$, the expert fits the red majority group, hence the classifier should attempt to fit the blue group with the rejector (black line) separating the groups.

5.1. Failure of Confidence Scores Method

The form of the Bayes solution in Proposition 5 above suggests a very natural approach: 1) learn a classifier minimizing the misclassification loss with the target and obtain confidence scores for predictions, 2) obtain confidence scores for expert agreement with the target, this can be done by learning a model where the target is whether the expert agrees with the task label and extracting confidence scores from this model (Raghu et al., 2019b), and finally 3) compare who between the classifier and the expert is more confident and accordingly defer. We refer to this as the confidence score method (Confidence), this approach leads to a consistent estimator for both the rejector and classifier and was proposed by Raghu et al. (2019a).

In fact this is the standard approach in rejection learning (Bartlett & Wegkamp, 2008; Ramaswamy et al., 2018; Ni et al., 2019), a host of different methods exist for estimating a classifier's confidence on new examples including trust scores (Jiang et al., 2018), Monte-Carlo dropout for neural networks (Gal & Ghahramani, 2016) among many others. However, the key pitfall of this method in the expert deferral setup it that it does not allow h to adapt to the expert's strengths and weaknesses. When we restrict our search space to a limited class of functions $\mathcal{H}$ and $\mathcal{R}$ this approach can easily fail. We now give a toy example where learning the classifier independently fails which motivates the need to jointly learn both the classifier and rejector.

Assume that there exists two sub-populations in the data denoted $A = 1$ and $A = 0$ where $\mathbb{P}(A = 1) \geq \mathbb{P}(A = 0)$ from which $X \in \mathbb{R}^d$ is generated from and conditional on the target and population, $X|(Y = y, A = 0)$ is normally distributed according to $\mathcal{N}(\mu_{y,0}, \Sigma)$ and $X|(Y = y, A = 1)$ consists of two clusters: cluster (1) is normally distributed but the means are not well separated and cluster (2) is only separable by a complex non-linear boundary; the data is illustrated in Figure 1. Finally we assume the expert to be able to perfectly classify group $A = 1$, on cluster (1) the expert is able to compute the complex nonlinear boundary and on cluster (2) the expert has side-information Z that allows him to separate the classes which is not possible from only X. We restrict our classifier and rejector to be $d-$dimensional hyperplanes. If we start by learning h, then the resulting hyperplane will try to minimize the average error across both groups, this will likely result into a hyperplane that separates neither group as the data is not linearly separable, especially on group $A = 1$. If we assume that the boundary between the groups is linear as shown, then we can achieve the error of the Bayes solution within our hypothesis space: the optimal behavior in this setting is clearly to have h fit group $A = 0$, note here the Bayes solution corresponds to a hyperplane via linear discriminant analysis for 2 classes on $A = 0$, and the rejector r separating the groups as illustrated in Figure 1. This example illustrates the complexities of this setting, due to model capacity there are significant gains to be achieved from adapting to the expert by focusing only group $A = 0$. Setting aside model capacity, the nonlinear boundary of cluster (1) is sample intensive to learn as we only have access to finite data. Finally, cluster (2) cannot be separated even with infinite data, the side information of the expert is needed, and so the hard task is to identify the region of cluster (2).

5.2. Inconsistency of Mixtures of Experts Loss

Note that the expert deferral setting considered here can be thought of as a hard mixture of two experts problem where one of the experts is fixed (Jordan & Jacobs, 1994; Shazeer et al., 2017; Madras et al., 2018). This observation motivates a natural mixture of experts type loss, let $g_y : \mathcal{X} \to \mathbb{R}$ for $y \in \mathcal{Y}$, $h(x) = \arg\max_{y \in \mathcal{Y}} g_y$, $r_i : \mathcal{X} \to \mathbb{R}$ for $i \in \{0, 1\}$ and $r(x) = \arg\max_{i \in \{0,1\}} r_i(x)$, the mixture of experts loss is defined as:

$$L_{mix}(\mathbf{g}, \mathbf{r}, x, y, m) = -\log\left(\frac{\exp(g_y(x))}{\sum_{y' \in \mathcal{Y}} \exp(g_{y'}(x))}\right) \cdot \tag{8}$$

$$\frac{\exp(r_0(x))}{\sum_{i \in \{0,1\}} \exp(r_i(x))} + \mathbb{I}_{m \neq y} \frac{\exp(r_1(x))}{\sum_{i \in \{0,1\}} \exp(r_i(x))}$$

The above loss extends Madras et al. (2018) approach to the multiclass setting. As the next proposition demonstrates, L_{mix} is in general *not* classification consistent.

Proposition 3. *L_{mix} is not a consistent surrogate loss function for L (3).*

Proof of proposition (3) can be found in Appendix C. In our experimental section we show how the mismatch between the model and expert loss and their actual errors arising from the inconsistency causes this method to learn the incorrect behavior.

5.3. Generalization Bound for Joint Learning

In this subsection we analyze the sample complexity to jointly learn a rejector and classifier. The goal is to find the minimizer of the empirical version of our system loss when our hypothesis space for h and r are $\mathcal{H}, \mathcal{R}$ respectively:

$$\hat{h}^*, \hat{r}^* = \arg \min_{h \in \mathcal{H}, r \in \mathcal{R}} L_{0-1}^S(h, r) \tag{9}$$

By going after the system loss directly, we can approximate the population minimizers h^*, r^* over $\mathcal{H} \times \mathcal{R}$ of L_{0-1} (3). The optimum h^* may not necessarily coincide with the optimal minimizer of the misclassification loss with the target which is why learning jointly is critical. We now give a generalization bound for our empirical minimization procedure for a binary target.

Theorem 2. *For any expert M and data distribution $\mathbf{P}$ over $\mathcal{X} \times \mathcal{Y}$, let $0 < \delta < \frac{1}{2}$, then with probability at least $1 - \delta$, the following holds for the empirical minimizers $(\hat{h}^*, \hat{r}^*)$:*

$$\begin{aligned} L_{0-1}(\hat{h}^*, \hat{r}^*) &\leq L_{0-1}(h^*, r^*) + \mathfrak{R}_n(\mathcal{H}) + \mathfrak{R}_n(\mathcal{R}) \\ &+ \mathfrak{R}_{n\mathbb{P}(M \neq Y)/2}(\mathcal{R}) + 2\sqrt{\frac{\log \frac{2}{\delta}}{2n}} \\ &+ \frac{\mathbb{P}(M \neq Y)}{2} \exp\left(-\frac{n\mathbb{P}(M \neq Y)}{8}\right) \end{aligned} \tag{10}$$

Proof of the above theorem can be found in Appendix C. We can see that the performance of our empirical minimizer is controlled by the Rademacher complexity $\mathfrak{R}_n(\mathcal{R})$ and $\mathfrak{R}_n(\mathcal{H})$ of both the classifier and rejector model classes and the error of the expert. Note that when $\mathbb{P}(M \neq Y) = 0$ we recover the bound proved in Theorem 1 (Cortes et al., 2016b) for rejection learning when $c = 0$; this gives evidence that deferring to an expert is a more sample intensive problem then rejection learning. Both our loss L_{CE} and the confidence scores approach lead to consistent estimators, however, as we will later show in our experiments, one differentiating factor will be that of sample complexity. We can already see in the bound (10), that we pay the complexity of the rejector and classifier model classes, however, our approach combines the rejector and classifier in one model to avoid these added costs.

6. Experiments

We provide code to reproduce our experiments [1]. Additional experimental details and results are left to Appendix B.

6.1. Synthetic Data

As a first toy example to showcase that our proposed loss L_{CE}^{α} is able to adapt to the underlying expert behavior, we perform experiments in a Gaussian mixture setup akin to the example in section 5. The covariate space is $\mathcal{X} = \mathbb{R}^d$ and target $\mathcal{Y} = \{0,1\}$, we assume that there exists two sub-populations in the data denoted $A = 1$ and $A = 0$. Furthermore, $X|(Y = y, A = a)$ is normally distributed according to $\mathcal{N}(\mu_{y,a}, \Sigma_{y,a})$. The expert follows the Bayes solution for group $A = 1$ which here corresponds to a hyperplane. Our hypothesis spaces $\mathcal{H}$ and $\mathcal{R}$ will be the set of all $d-$dimensional hyperplanes.

Setup: We perform 200 trials where on each trial we generate: random group proportions $\mathbb{P}(A = 1) \sim U(0,1)$ fixing $\mathbb{P}(Y = 1|A = a) = 0.5$, random means and variances for each Gaussian component $X|Y = y, A = a \sim \mathcal{N}(\mu_{y,a}, \Sigma_{y,a})$ where $\mu_{y,a} \sim U(0,10)^d$ and similarly for the diagonal components of $\Sigma_{y,a}(i,i) \sim U(0,10)$ keeping non-diagonal components 0 with dimension $d = 10$; we generate in total 1000 samples each for training and testing. We compare against oracle behavior and two baselines: 1) An oracle baseline (Oracle) that trains only on $A = 0$ data and trains the rejector to separate the groups with knowledge of group labels and 2) the confidence score baseline (Confidence) that trains a linear model on all the data and then trains a different linear model on all the data where labels are the expert's agreement with the target and finally compares which of the two is more confident according to the probabilities assigned by the corresponding models and 3) our implementation of the approach in (Madras et al., 2018) (MixOfExp).

Results: We train a multiclass logistic regression model with our loss L_{CE}^{α} with $\alpha \in \{0, 0.5, 1\}$ and record in table 1 the difference in accuracy between our method and baselines for the best performing α. We can see that our method with $\alpha = 0$ outperforms the confidence baseline by 6.39 on average in classification accuracy and matches the oracle method with 0.22 positive difference which shows the success of our method.

Table 1. Average difference in accuracy for our method compared to the baselines and a 95% confidence interval for the average for the synthetic data experiment.

DIFFERENCE IN SYSTEM ACCURACY	AVERAGE	95% INTERVAL
L_{CE}^{0}-CONFIDENCE (RAGHU ET AL., 2019A)	6.39	[3.71,9.06]
L_{CE}^{0}-ORACLE	0.22	[-1.71,2.15]
L_{CE}^{0}- MIXOFEXP (MADRAS ET AL., 2018)	2.01	[0.14,4.06]

6.2. CIFAR-10

As our first real data experimental evaluation we conduct experiments on the celebrated CIFAR-10 image classification dataset (Krizhevsky et al., 2009) consisting of 32×32 color images drawn from 10 classes split into 50,000 train and 10,000 test images.

Synthetic Expert. We simulate multiple synthetic experts of varying competence in the following way: let $k \in [10]$, then if the image belongs to the first k classes the expert predicts perfectly, otherwise the expert predicts uniformly over all classes. The classifier and expert costs are assumed to be the misclassification costs.

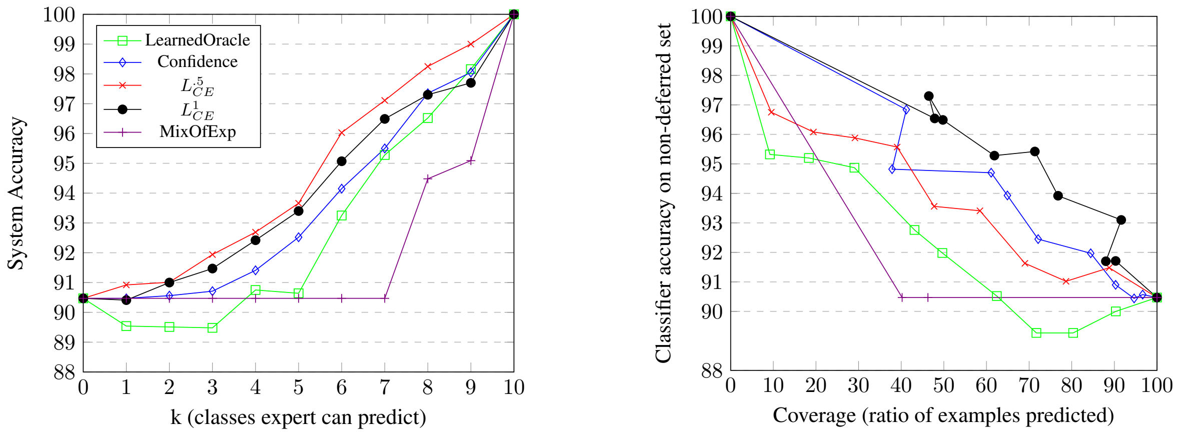
Base Network. Our base network for classification will be the Wide Residual Networks (WideResNets) (Zagoruyko & Komodakis, 2016) which with data augmentation and hyperparameter tuning can achieve a 96.2% test accuracy. Since our goal is not to achieve better accuracies but to show the merit of our approach for a given fixed model, we disadvantage the model by not using data augmentation and a smaller network size. The WideResNet with 28 layers minimizing the cross-entropy loss achieves 90.47% test accuracy with training until fitting the data in 200 epochs; this will be our benchmark model. We use SGD with momentum and a cosine annealing learning rate schedule.

Proposed Approach: Following section 4, we parameterize h and r (specifically $g_\perp$) by a WideResNet with 11 output units where the first 10 units represent h and the $11'th$ unit is $g_\perp$ and minimize the proposed surrogate L_{CE}^{α} (6). We also experimented with having h be a WideResNet with 10 output units and $g_\perp$ a WideResNet with a single output unit and observed identical results. We show results for $\alpha \in \{0.5, 1\}$.

Baselines: We compare against three baselines. The first baseline trains the rejector to recognize if the image is in the first k classes and accordingly defers, we call this baseline "LearnedOracle"; this rejector is a learned implementation of what the optimal rejector should do. The second baseline is the confidence score method (Raghu et al., 2019a) and the third is the mixture-of-experts loss of (Madras et al., 2018), details of the implementation of this final baseline are left to Appendix B.5.

Results. In figure 2a we plot the accuracy of the combined algorithm and expert system versus k, the number of classes the expert can predict perfectly. We can see that the model trained with $L_{CE}^{0.5}$ and L_{CE}^{1} outperforms the baselines by 1.01% on average for the confidence score baseline and by 1.94 on average for LearnedOracle. To look more closely at the behavior of our method, we plot in figure 2b the

[1] https://github.com/clinicalml/learn-to-defer



(a)

(b)

Figure 2. Left figure shows overall system accuracy of our method and baselines (k is the number of classes the expert can predict) and right figure compares the accuracy on the non-deferred examples versus the coverage for every k

accuracy on the non-deferred examples versus the coverage, the fraction of the examples non-deferred, for each k. We can see that that the model trained with L_{CE}^1 dominates all other baselines giving better coverage and accuracy for the classifier's predictions. This gives evidence that our loss allows the model to only predict when it is highly confident.

Why do we outperform the baselines?
1) *Sample complexity*: The Confidence baseline (Raghu et al., 2019a) requires training two networks while ours only requires one, when data is limited our approach gives significant improvements in comparison. We experiment with increasing training set sizes while keeping the test set fixed and training our model with L_{CE}^1 and the Confidence baseline. With expert $k = 5$, when data is limited our approach massively improves on the baseline, for example with 2000 training points, Confidence achieves 62.33% accuracy while our method achieves 70.12%, a 7.89 point increase.
2) *Taking into consideration both expert and model confidence*: the LearnedOracle baseline ignores model confidence entirely and only focuses on the region where the expert is correct. While this is the behavior of the Bayes classifier in this setup, when dealing with a limited model class and limited data, this no longer is the correct behavior. For this reason, our model outperforms the LearnedOracle baseline.
3) *Consistency*: the mixtures of experts loss of (Madras et al., 2018) fails in this setup and learns never to defer. The reason is that when training, the loss of the classifier will converge to zero and validation classifier accuracy will still improve in the mean-time, however the loss of the expert remains constant, thus we never defer.

6.3. CIFAR10H and Limited Expert Data

In the following experiments we assume access to fully labeled data $S_l = \{(x_i, y_i, m_i)\}_{i=1}^m$ and data without expert labels $S_u = \{(x_i, y_i)\}_{i=m+1}^n$. The goal is to learn a classifier h and rejector r from the two datasets S_l and S_u.
Data. To experiment in settings where we have limited expert data, we use the dataset `CIFAR10H` (Peterson et al., 2019) initially developed to improve model robustness. `CIFAR10H` contains for each data point in the CIFAR-10 test set fifty crowdworker annotations recorded as counts for each of the 10 classes. The training set of CIFAR-10 will constitute S_u, and we randomly split the test set in half where one half constitutes S_l and the other is for testing; we randomize the splitting over 10 trials.
Expert. We simulate the behavior of an average human annotator by sampling from the class counts for each data point. The performance of our simulated expert has an average classification accuracy of 95.22 with a standard deviation of 0.18 over 100 runs. The performance of the expert is non uniform over the classes, for example on the class *cat* the expert has 91.0% accuracy while on *horse* a 97.8% accuracy.
Proposed Approach. Our method will be to impute expert disagreement labels $\mathbb{I}_{y \neq m}$ on S_u by learning a model that predicts whether the expert will err and obtain an imputed dataset $\hat{S}_u$. We train using our loss L_{CE} on $\hat{S}_u \cup S_l$; we refer to our method as "L_{CE} impute".
Results. We compare against a confidence score baseline where we train a classifier on S_u and then model the expert on S_l. Results are shown in table 2 and we can see that our method outperforms the confidence method by 1.2 points on system accuracy and an impressive 3.1 on data points where the classifier has to predict. To show the effect of imputing expert labels on S_u, we train first our model using

Table 2. Comparing our proposed methods on CIFAR10H and a baseline based on confidence scores recording system accuracy, coverage and classifier accuracy on non-deferred examples.

METHOD	SYSTEM	COVERAGE	CLASSIFIER
L_{CE} IMPUTE	**96.29**±0.25	51.67±1.46	**99.2** ± 0.08
L_{CE} 2-STEP	96.03±0.21	60.81±0.87	98.11 ± 0.22
CONFIDENCE (RAGHU ET AL., 2019A)	95.09±0.40	**79.48**±5.93	96.09 ± 0.42

L_{CE} on S_u and then fine tune to learn deferral on S_l, we refer to this as "L_{CE} 2-step".

6.4. CheXpert

Task. CheXpert is a large chest radiograph dataset that contains over 224 thousand images of 65,240 patients automatically labeled for the presence of 14 observations using radiology reports (Irvin et al., 2019). We focus here on the detection of only the 5 observations that make up the "competition tasks" (Irvin et al., 2019): Atelectasis, Cardiomegaly, Consolidation, Edema, and Pleural Effusion. This is a multi-task problem, we have 5 separate binary tasks, we will learn to defer on an individual task basis.

Expert. We create a simulated expert as follows: if the chest X-ray contains support devices (the presence of support devices is part of the label) then the expert is correct with probability p on all tasks independently and if the X-ray does not contain support devices, then the expert is correct with probability q.

Data. We use the downsampled resolution version of CheXpert (Irvin et al., 2019) and split the training data set with an 80-10-10 split on a patient basis for training, validation and testing respectively, no patients are shared among the splits. Images are normalized and resized to be compatible with pre-trained ImageNet models, we use data augmentation in the form of random resized crops, horizontal flips and random rotations of up to $15°$ while training.

Baselines. We implement two baselines: a threshold confidence baseline that learns a threshold to maximize system AU-ROC on just the confidence of the classifier model to defer (ModelConfidence), this is the post-hoc thresholding method in (Madras et al., 2018), and the Confidence baseline (Raghu et al., 2019a). We use temperature scaling (Guo et al., 2017) to ensure calibration of all baselines on the validation set.

Model. Following (Irvin et al., 2019), we use the DenseNet121 architecture for our model with pre-trained weights on ImageNet, the loss for the baseline models is the average of the binary cross entropy for each of the tasks. We train the baseline models using Adam for 4 epochs. For our approach we train for 3 epochs using the cross entropy loss and then train for one epoch using L_{CE}^{α} with α chosen to maximize the area under the receiver operating characteristic curve (AU-ROC) of the combined system on the validation set for each of the 5 tasks (each task is treated separately).

Experimental setup. In a clinical setting there might be a cost associated to querying a radiologist, this then imposes a constraint on how often we can query the radiologist i.e. our model's coverage . We constrain our method and the baselines to achieve $c\%$ coverage for $c \in [100]$ to simulate the spectrum between complete automation and none.

We achieve this for our method by first sorting the test set based on $g_\perp(x) - \max(g_0(x), g_1(x)) := q(x)$ across all patients x in the test set, then to achieve coverage c, we define $\tau = q(x_c)$ where $q(x_c)$ is the c'th percentile of the outputs $q(x)$, then we let $r(x) = 1 \iff q(x) \geq \tau$. The definition of τ ensures that we obtain exactly $c\%$ coverage. We do the this similarly for ModelConfidence and Confidence.

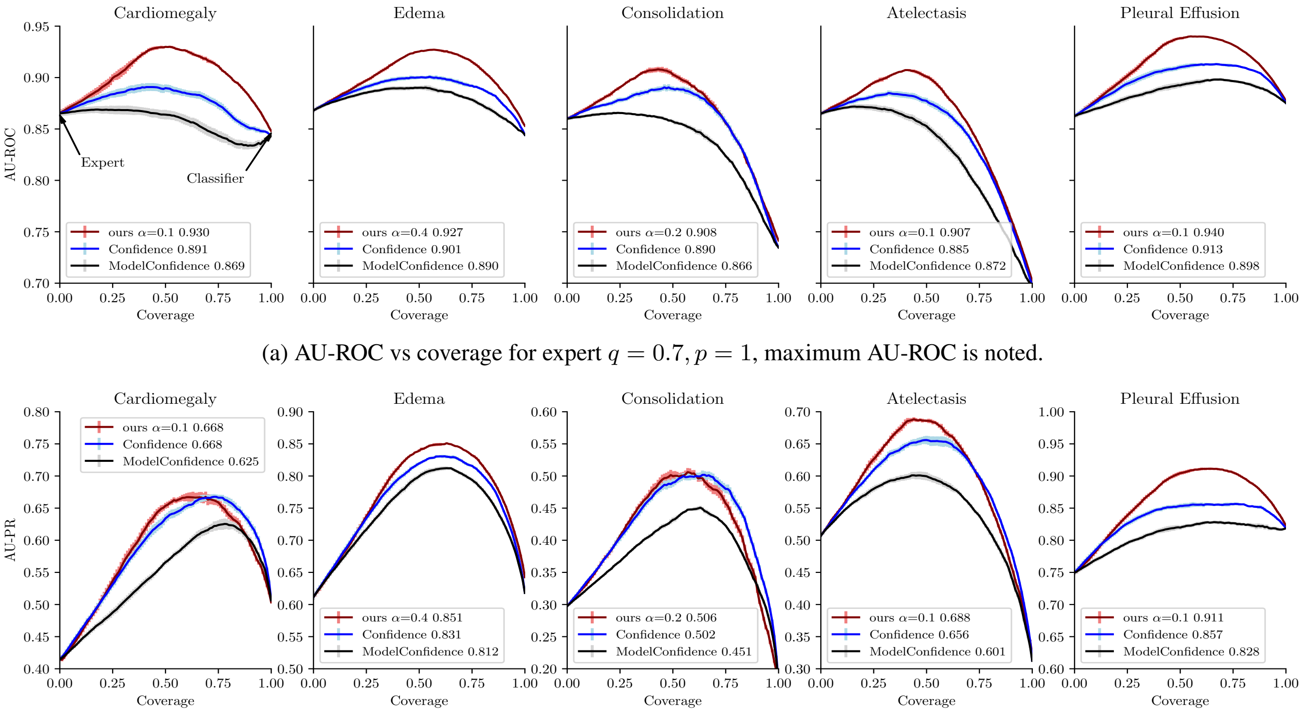
Results. In Figure 3a we plot the overall system (expert and algorithm combined) AU-ROC for each desired coverage for the methods and in Figure 3b we plot the overall system area under the precision-recall curve (AU-PR) versus the coverage; this is for the expert with $q = 0.7$ and $p = 1$. We can see that the curve for our method dominates the baselines over the entire coverage range for both AU-ROC and AU-PR, moreover the curves are concave and we can achieve higher performance by combining expert and algorithm than using both separately. Our method is able to achieve a higher maximum AU-ROC and AU-PR than both baselines: the difference between the maximum attainable AU-ROC of our method and Confidence is 0.039, 0.026, 0.018, 0.022 and 0.027 respectively for each of the five tasks.

6.5. Hate Speech and Offensive Language Detection

We conduct experiments on the dataset created by Davidson et al. (2017) consisting of 24,783 tweets annotated as hate speech, offensive language or neither.

Expert. We create a synthetic expert that has differing error rates according to the demographic of the tweet's author, using the probabilistic language model of (Blodgett et al., 2016) we predict that a tweet is in African-American English (AAE) if the probability predicted by the model is higher than 0.5. Our expert model is as follows: if the tweet is in AAE then with probability p we predict the correct label and otherwise predict uniformly at random. On the other hand if the tweet is not in AAE, we predict with probability q the correct label. We experiment with 3 different expert probabilities for p and q: 1) a fair expert with $\{p = 0.9, q = 0.9\}$, 2) a biased expert towards AAE tweets $\{p = 0.75, q = 0.9\}$ and 3) a biased expert towards non AAE tweets $\{p = 0.9, q = 0.75\}$.

Our Approach. For our model we use the CNN developed in (Kim, 2014) for text classification with 100 dimensional Glove embeddings (Pennington et al., 2014) and 300 filters



(a) AU-ROC vs coverage for expert $q = 0.7, p = 1$, maximum AU-ROC is noted.

(b) AU-PR vs coverage for expert $q = 0.7, p = 1$, maximum AU-PR is noted.

Figure 3. Plot of AU-ROC of the ROC curve (a) for each level of coverage and of the AU-PR (AP) (b) for each of the 5 tasks comparing our method with the baselines on the training derived test set for the toy expert with $q = 0.7, p = 1$. We report the maximum AU-ROC and AU-PR achieved on each task, error bars are standard deviations derived from 10 runs (averaging over the expert's randomness).

of sizes $\{3, 4, 5\}$ using dropout. This CNN achieves a 89.5% average accuracy on the classification task, comparable to the 91% achieved by (Davidson et al., 2017) with a feature heavy linear model.

We randomly split the dataset with a $60, 10, 30\%$ split into a training, validation and test set respectively; we repeat the experiments for 5 random splits. We used a grid search over the validation set to find α.

Results. We compare against two baselines: the first is Confidence, the second is an oracle baseline that trains first a model on the classification task and then implements the Bayes rejector $r^B(x)$ equipped with the knowledge of p, q and the tweet's demographic group. Both our model trained with L_{CE}^1 and the confidence score baseline achieve similar accuracy and coverage with the oracle baseline performing only slightly better across the three experts. For the AAE biased expert, our model trained with L_{CE}^1 achieves 92.91±0.17 system accuracy, Confidence 92.42±0.40 and Oracle 93.22±0.11. This suggests that both approaches are performing optimally in this setting.

Bias. A major concern in this setting is whether the end to end system consisting of the classifier and expert is discriminatory. We define the discrimination of a predictor as the difference in the false positive rates of AAE tweets versus non AAE tweets where false positives indicate tweets that were flagged as hate speech or offensive when they were not. Surprisingly, the confidence score baseline with the fair expert doubles the discrimination of the overall system compared to the classifier acting on it's own: the classifier has a discrimination of 0.226 , the fair expert 0.03 while the confidence score baseline has a discrimination of 0.449. This again reiterates the established fact that fairness does not compose (Dwork & Ilvento, 2018). In fact, the end-to-end system can be less discriminatory even if the individual components are more discriminatory, for the second expert that has higher error rates on non AAE tweets with discrimination of 0.084, the discrimination of the confidence score method reduces to 0.151.

7. Conclusion

In this work we explored a framework where the learning model can choose to defer to an expert or predict. We analyzed the framework theoretically and proposed a novel surrogate loss via a reduction to multiclass cost sensitive learning. We showcased on image and text classifications tasks the empirical benefits of our method compared to the literature. We hope that our method will inspire machine learning practitioners to integrate downstream decision makers into their learning algorithms.

Acknowledgements

This work was supported by NSF CAREER award #1350965.

References

Bansal, G., Nushi, B., Kamar, E., Weld, D. S., Lasecki, W. S., and Horvitz, E. Updates in human-ai teams: Understanding and addressing the performance/compatibility tradeoff. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 2429–2437, 2019.

Bansal, G., Nushi, B., Kamar, E., Horvitz, E., and Weld, D. S. Optimizing ai for teamwork. *arXiv preprint arXiv:2004.13102*, 2020.

Bartlett, P. L. and Wegkamp, M. H. Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, 9(Aug):1823–1840, 2008.

Bartlett, P. L., Jordan, M. I., and McAuliffe, J. D. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006.

Blodgett, S. L., Green, L., and O'Connor, B. Demographic dialectal variation in social media: A case study of African-American English. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1119–1130, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1120. URL https://www.aclweb.org/anthology/D16-1120.

Canetti, R., Cohen, A., Dikkala, N., Ramnarayan, G., Scheffler, S., and Smith, A. From soft classifiers to hard decisions: How fair can we be? In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 309–318. ACM, 2019.

Chen, M., Gummadi, R., Harris, C., and Schuurmans, D. Surrogate objectives for batch policy optimization in one-step decision making. In *Advances in Neural Information Processing Systems*, pp. 8825–8835, 2019.

Chow, C. On optimum recognition error and reject tradeoff. *IEEE Transactions on information theory*, 16(1):41–46, 1970.

Cortes, C., DeSalvo, G., and Mohri, M. Boosting with abstention. In *Advances in Neural Information Processing Systems*, pp. 1660–1668, 2016a.

Cortes, C., DeSalvo, G., and Mohri, M. Learning with rejection. In *International Conference on Algorithmic Learning Theory*, pp. 67–82. Springer, 2016b.

Davidson, T., Warmsley, D., Macy, M., and Weber, I. Automated hate speech detection and the problem of offensive language. In *Eleventh international aaai conference on web and social media*, 2017.

De, A., Koley, P., Ganguly, N., and Gomez-Rodriguez, M. Regression under human assistance. *arXiv preprint arXiv:1909.02963*, 2019.

DeSalvo, G., Mohri, M., and Syed, U. Learning with deep cascades. In *International Conference on Algorithmic Learning Theory*, pp. 254–269. Springer, 2015.

Dwork, C. and Ilvento, C. Fairness under composition. *arXiv preprint arXiv:1806.06122*, 2018.

El-Yaniv, R. and Wiener, Y. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(May):1605–1641, 2010.

Friedman, J., Hastie, T., and Tibshirani, R. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

Futoma, J., Hariharan, S., Heller, K., Sendak, M., Brajer, N., Clement, M., Bedoya, A., and O'Brien, C. An improved multi-output gaussian process rnn with real-time validation for early sepsis detection. In *Machine Learning for Healthcare Conference*, pp. 243–254, 2017.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059, 2016.

Geifman, Y. and El-Yaniv, R. Selective classification for deep neural networks. In *Advances in neural information processing systems*, pp. 4878–4887, 2017.

Geifman, Y. and El-Yaniv, R. Selectivenet: A deep neural network with an integrated reject option. *arXiv preprint arXiv:1901.09192*, 2019.

Green, B. and Chen, Y. Disparate interactions: An algorithm-in-the-loop analysis of fairness in risk assessments. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 90–99. ACM, 2019a.

Green, B. and Chen, Y. The principles and limits of algorithm-in-the-loop decision making. *Proceedings of the ACM on Human-Computer Interaction*, 3(CSCW): 1–24, 2019b.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1321–1330. JMLR. org, 2017.

Hamid, K., Asif, A., Abbasi, W., Sabih, D., et al. Machine learning with abstention for automated liver disease diagnosis. In *2017 International Conference on Frontiers of Information Technology (FIT)*, pp. 356–361. IEEE, 2017.

Irvin, J., Rajpurkar, P., Ko, M., Yu, Y., Ciurea-Ilcus, S., Chute, C., Marklund, H., Haghgoo, B., Ball, R., Shpanskaya, K., et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 590–597, 2019.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Jhaver, S., Birman, I., Gilbert, E., and Bruckman, A. Human-machine collaboration for content regulation: The case of reddit automoderator. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 26(5):1–35, 2019.

Jiang, H., Kim, B., Guan, M., and Gupta, M. To trust or not to trust a classifier. In *Advances in neural information processing systems*, pp. 5541–5552, 2018.

Jordan, M. I. and Jacobs, R. A. Hierarchical mixtures of experts and the em algorithm. *Neural computation*, 6(2): 181–214, 1994.

Kim, Y. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

Kolesnikov, A., Beyer, L., Zhai, X., Puigcerver, J., Yung, J., Gelly, S., and Houlsby, N. Large scale learning of general visual representations for transfer. *arXiv preprint arXiv:1912.11370*, 2019.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. *Citeseer*, 2009.

Link, D., Hellingrath, B., and Ling, J. A human-is-the-loop approach for semi-automated content moderation. In *ISCRAM*, 2016.

Loshchilov, I. and Hutter, F. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Maddison, C. J., Mnih, A., and Teh, Y. W. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Madras, D., Pitassi, T., and Zemel, R. Predict responsibly: Improving fairness and accuracy by learning to defer. In *Advances in Neural Information Processing Systems*, pp. 6150–6160, 2018.

Meresht, V. B., De, A., Singla, A., and Gomez-Rodriguez, M. Learning to switch between machines and humans. *arXiv preprint arXiv:2002.04258*, 2020.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of machine learning*. MIT press, 2018.

Ni, C., Charoenphakdee, N., Honda, J., and Sugiyama, M. On possibility and impossibility of multiclass classification with rejection. *arXiv preprint arXiv:1901.10655*, 2019.

Pennington, J., Socher, R., and Manning, C. D. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

Peterson, J. C., Battleday, R. M., Griffiths, T. L., and Russakovsky, O. Human uncertainty makes classification more robust. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9617–9626, 2019.

Raghu, M., Blumer, K., Corrado, G., Kleinberg, J., Obermeyer, Z., and Mullainathan, S. The algorithmic automation problem: Prediction, triage, and human effort. *arXiv preprint arXiv:1903.12220*, 2019a.

Raghu, M., Blumer, K., Sayres, R., Obermeyer, Z., Kleinberg, B., Mullainathan, S., and Kleinberg, J. Direct uncertainty prediction for medical second opinions. In *International Conference on Machine Learning*, pp. 5281–5290, 2019b.

Ramaswamy, H. G., Tewari, A., Agarwal, S., et al. Consistent algorithms for multiclass classification with an abstain option. *Electronic Journal of Statistics*, 12(1): 530–554, 2018.

Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., and Dean, J. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*, 2017.

Vincent, J. Ai won't relieve the misery of facebook's human moderators. `https://www.theverge.com/2019/2/27/18242724/facebook-moderation-ai-artificial-intelligence-platforms`, February 2019.

Wilder, B., Horvitz, E., and Kamar, E. Learning to complement humans. *arXiv preprint arXiv:2005.00582*, 2020.

Zagoruyko, S. and Komodakis, N. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

Zhang, C. and Chaudhuri, K. Active learning from weak and strong labelers. In *Advances in Neural Information Processing Systems*, pp. 703–711, 2015.

Ziyin, L., Wang, Z., Liang, P. P., Salakhutdinov, R., Morency, L.-P., and Ueda, M. Deep gamblers: Learning to abstain with portfolio theory. *arXiv preprint arXiv:1907.00208*, 2019.