Rejector on Top of Baseline for Learning to Defer to an Expert

A Project Report
Submitted in Partial Fulfillment of Requirements
For the Degree of
Bachelor of Technology

by

Aditya Goel (2020MCB1259)
Rohan Kumar (2020MCB1247)

Department of Mathematics and Computing
Indian Institute of Technology Ropar
Rupnagar 140001, India

May 2024

Abstract

Humans and machines have different strengths. Machines excel at processing large amounts of data quickly and accurately, while humans excel at contextual understanding and intuition, and nuanced decision-making. By combining these strengths, we can accomplish tasks more effectively and efficiently. In practical circumstances, learning algorithms frequently work in tandem with expert decision makers; nevertheless, this aspect is frequently overlooked throughout the algorithm's design process. This research investigates the learning of predictors that have the ability to choose to defer a choice to a downstream expert. We provide an approach based on learning a rejector, given only samples of the expert's decisions. We demonstrate our method's efficacy on a range of experimental tasks.

Acknowledgements

We want to express our deepest gratitude to our advisor, Dr. Shweta Jain, for her invaluable guidance, support, and mentorship throughout the course of our thesis. Her expertise and insights have been instrumental in shaping our work, and we are truly indebted to her for the opportunity to work under her supervision.

We would like to acknowledge the kind support of our institute, IIT Ropar, for providing us access to the High-Performance Computing facility, which has enabled us to perform complex simulations and computations.

We are also grateful to our advisor for providing us access to the GPU from her lab, which has significantly accelerated our work and made our research possible.

Contents

Chapter 1

Introduction

The use of Artificial Intelligence has advanced at a tremendous rate in the past few years. It is used in almost all sectors such as healthcare, finance, education, job recruitment, cyber-security etc. To use the models effectively, we need them to train on huge quantities of data. Also, we need them to be accurate and predict responsibly - the model should predict only if its predictions are reliably aligned with the system's objectives.

To ensure this, the idea of human assisted AI was introduced. The human and the machine work as a team and the human labels help the machine model to get better at prediction. However, we want the cost of the model to be minimized as well. So we categorize human assistance to be of high cost. This trains the model to become accurate while using as less human help as possible.

There are many possibile ways in how we can train such models. One such method is to train both the defer model and machine model together considering them as one. We try to build a rejector model independently from the baseline model in order to remove the cost of back propagation to update parameters of the baseline model. We try to show the effectiveness of our approach by running experiments on CIFAR-10 and showing that our results resemble the case where the whole model is being trained.

Chapter 2

Related Works

Madras et al.[3] formulated a framework of adaptive rejection learning, also known as *learning to defer* where the model works adaptively with the decision-maker. The model (machine learning model) and the defer model (whether to defer to the human or not) re trained together. In this approach the model is adapted to the human decision maker. Mozannar et al.[4] demonstrated *learning to defer* for multiclass datasets, notably on the CIFAR-10 dataset, which is also the dataset used in this paper.

Previous works have also explored combining results from both humans and the machine learning, notably Kerrigan et al.[2] proposed a framework for combining predictions using instance level confidence from a model and class level information from a human.

All the above work does not take into account the cost of human predictions. Training of both the model and defer model would require a lot of computation. Another factor which needs to be considered is the existence and growth of multiple machine learning models which are already achieving extremely high accuracies and are only getting better and better.

Learning with a reject option, *rejection learning*, has long been studied starting with C. Chow [1] 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. We aim to learn a more complex rejector so that we obtain the benefits of not having to train the machine learning model and still obtain comparable results.

Chapter 3

Problem Formulation

We will be working on the CIFAR-10 dataset in which our aim is to identify the correct class of the given image i.e., we are interested in predicting a target $y \in Y = \{1, ..., K\}$ based on the given $x \in X$. We assume that we have an access to an expert M who may have some extra information on the given x that helps him classify to the correct class. We introduce a confidence based deferring wherein we decide to defer if the machine is not confident in its output. Let us represent the defer module by r. So, $r(x) = 0$ when we decide not to defer and $r(x) = 1$ when we hand over the decision to an expert. Let us denote the probability vector which is given as output of the baseline model as $[Y]$ and the class predicted by the baseline model as $\hat{Y}$. Then for given x, our aim is to learn the function $r : [y] \longrightarrow \{0, 1\}$. We can define the 0-1 loss function for our method as follows:

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

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 $M \mid (X = x_i, Y = y_i)$.

Chapter 4

Methodology

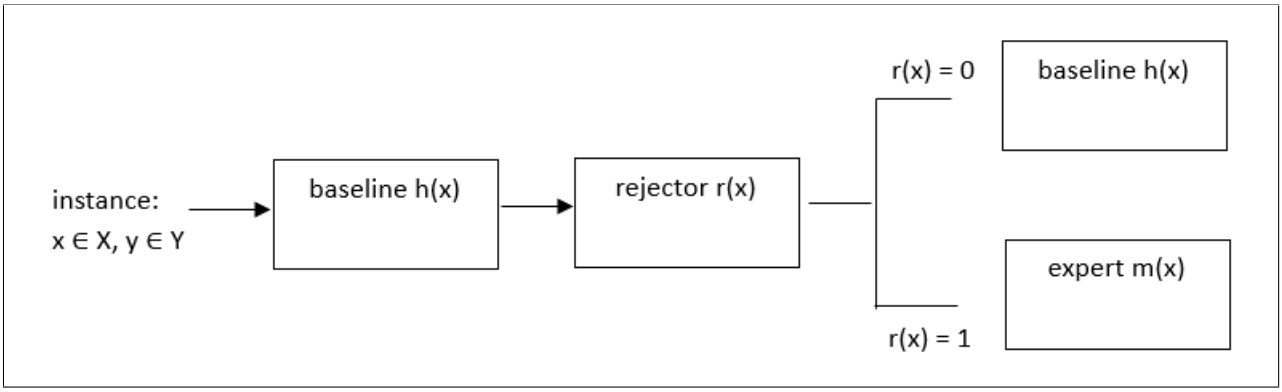


Figure 4.1: The expert deferral pipeline

Figure 4.1 represents the basic pipeline for our model and Algorithm 1 gives the pseudo code for the same. Assume that we have a baseline model which is not fully accurate on

Algorithm 1: Algorithm for Deferred Module

Input: Instance x, Baseline m, Expert h, Threshold t
Output: Predicted Label y
Let $h(x) = [c_1, c_2, ..., c_k]$ and $c_{max} \longleftarrow max_i\ c_i$
if $c_{max} < t$ **then**
  $\hat{y} = AskHuman()$;
else
  $\hat{y} = argmax_i\ c_i$
end

the training data. For that, we will be able to train the defer model and get results which is also demonstrated in section 5.5.

However, we want to compare our results from Mozannar et al. [4] in which baseline model trained for 200 epochs is used. This model is fully accurate on the training dataset. If we try to use the same to train an independent defer model on top, we will run into an issue. The baseline model will always generate correct predictions and when we try to train the parameters of the defer model, we won't be able to as the loss would be nil.

To solve this issue, we take advantage of the confidence of the machine. We introduce a threshold value specific to the dataset based upon the observations of the machine probabilities. Whenever the probability of the class predicted by the baseline falls short of threshold, we decide to defer to the expert. This introduces some defers in the target vector while training the neural net which provides us a well trained defer module.

Chapter 5

Experimentation and Results

5.1 CIFAR-10

For our real data experimental evaluation we conduct experiments on the CIFAR-10 image classification dataset consisting of 32x32 color images drawn from 10 classes split into 50,000 train and 10,000 test images wherein each class contains 6,000 images. From the set of train images, we use 90% as training data and remaining 10% as validation data. Images are present in a batch size of 128 per batch.

5.2 Synthetic Expert

For the human to be used for simulation purposes, we build up a synthetic expert in the following way. We define a value $k \in [10]$ such that the expert is fully accurate if the image belongs to first k classes of the dataset. If not, then expert predicts uniformly over all given 10 classes. Cost of expert is taken as misclassification cost itself.

5.3 Baseline Model

The baseline used in our approach will be Wide Residual Networks. WideResNet can achieve a test accuracy of 96.2% with hyperparamter tuning. But we purposely disadvantage the model by reducing the layers to 28 which achieves a test accuracy of 90.41% with training for 200 epochs on the training dataset, so we can compare the results with Mozannar et al.[4] .

From figure 5.1, we can see that the baseline model becomes fully accurate on the training data after 200 epochs. This will create a problem for our approach since we would not be able to train our defer model. This is explained in the further sections.

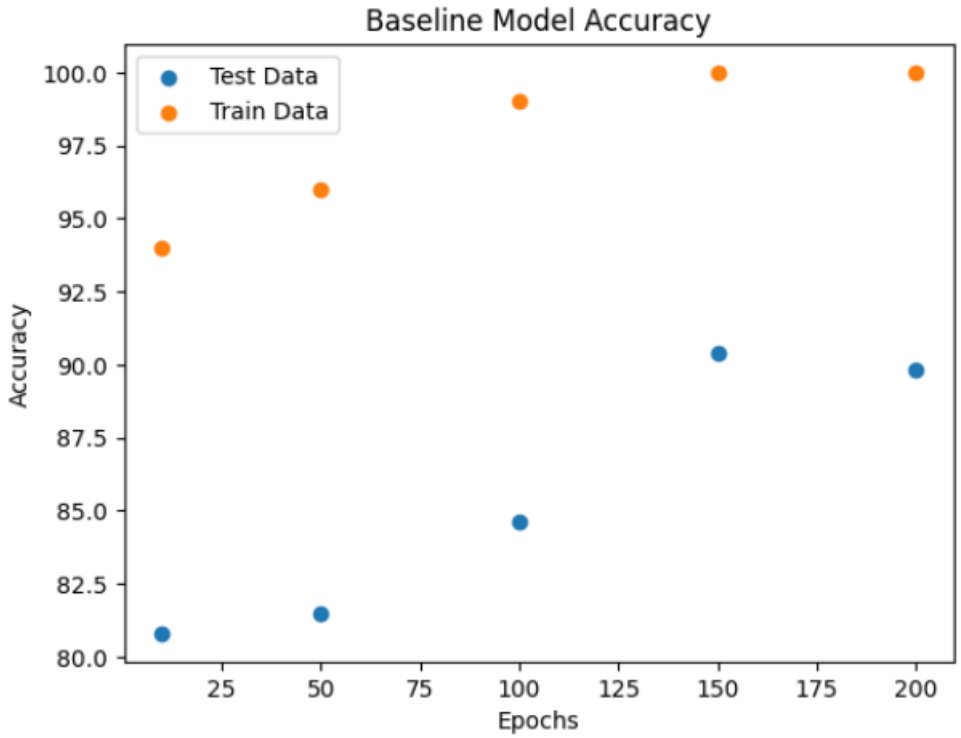


Figure 5.1: WideResNet Accuracy

5.4 Proposed Approach

As described in the methodology, we take an image x as input to the baseline model which outputs a probability vector $[Y]$ which is a vector containing 10 elements. An element Y_i denoted the probability of x being of class i. Now this vector $[Y]$ is taken as input to the defer model which returns a value between 0 and 1. Based on the output, we decide to defer to an expert or not.

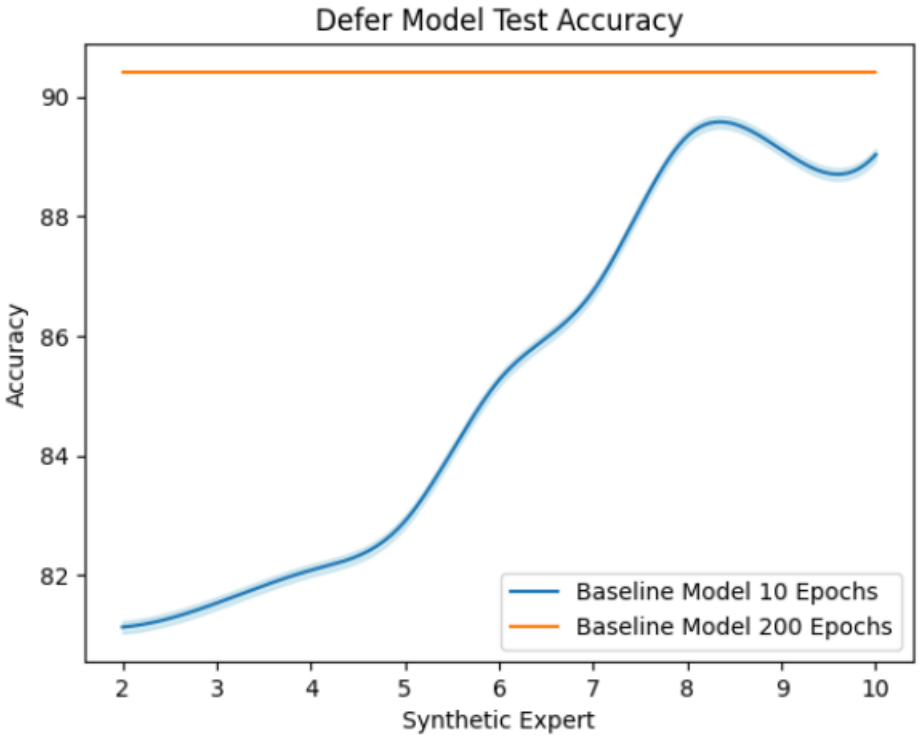
We talked about the problem arising due to high accuracy of the baseline model. Due to high accuracy, the machine will always give the right answer on training the defer model and hence, our model will learn to never defer. This creates a problem for us as the defer model does no improvement in the final model and the final test accuracy is same as the test accuracy of the baseline.

To rectify it, we introduced confidence based deferring. We want the model to defer whenever the machine is not confident in its output. To achieve this we can set a threshold on the probability of the predicted class of the baseline, below which if the human output is correct, then the target label is set to 1. We set the threshold to 0.999 for our experiments.

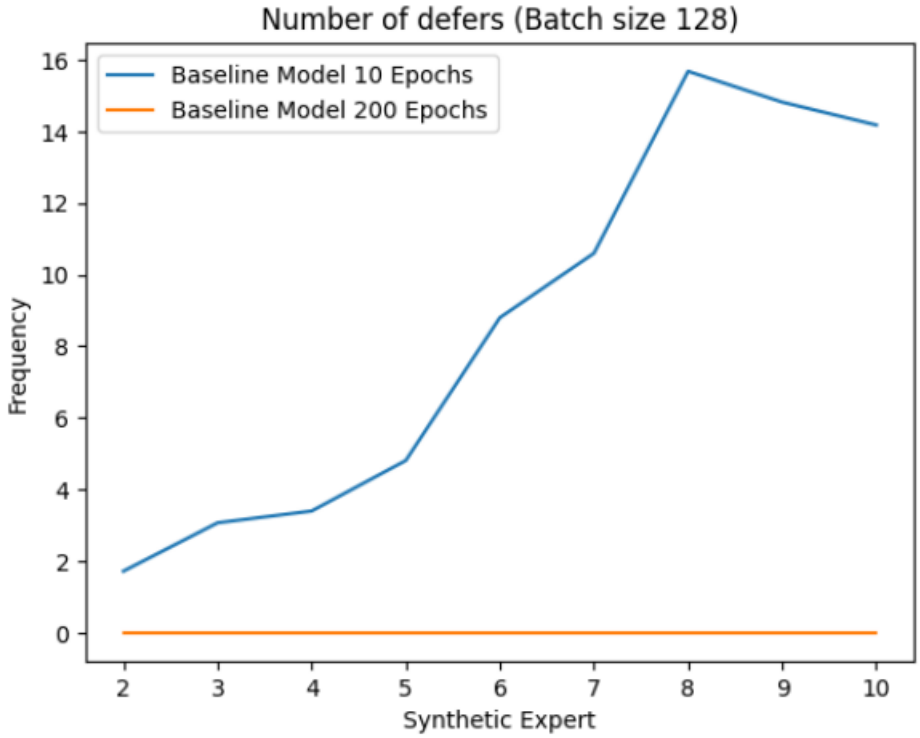
5.5 Results

Let us first look at the results obtained if we consider the baseline model trained for 10 epochs only and then for 200 epochs.

If we want to compare results with the case where the baseline model and defer model

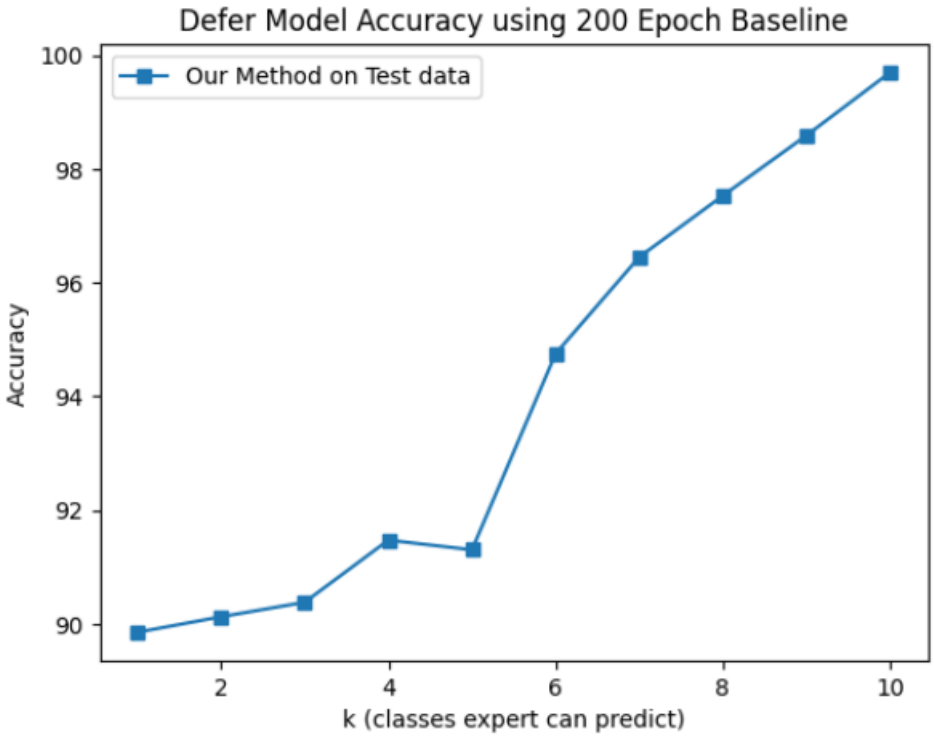


(a)

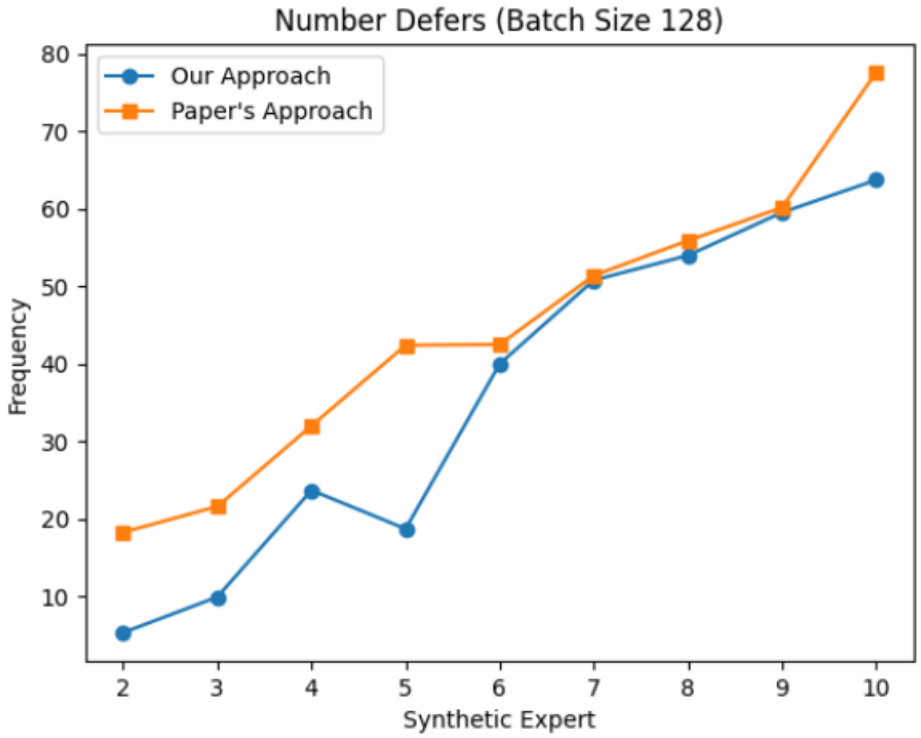


(b)

Figure 5.2: Using Baseline model trained for 10 Epochs



(a)



(b)

Figure 5.3: Using Baseline model trained for 200 Epochs

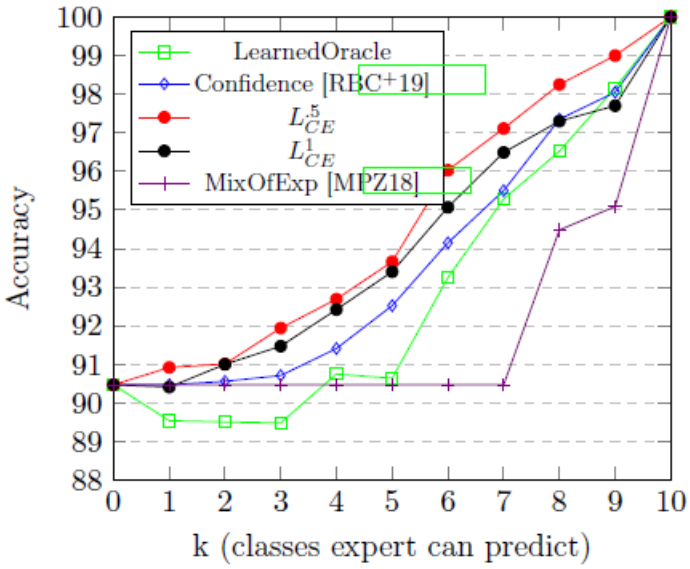


Figure 5.4: Mozannar Et Al. Results

are trained together, we can refer to Mozannar et al.[4] in figure 5.4.

From figure 5.3a and figure 5.4, we can conclude that our approach provides similar results i.e. training defer model separately is sufficient and saves a lot of computation cost.

Chapter 6

Future Works

For this work, we propose several avenues for future research and development:

1. **Label Combination :** Till now we have always taken the final class of the image as being one provided by the baseline or the human in case of defer. But consider this, when we are deferring to the human expert for the assistance, we also have the information about the machine prediction. We are not utilising this information. To use this, one can combine the labels using methods mentioned in Kerrigan et al.[2].

2. **Calibration :** For supervised learning tasks such as classification, it becomes crucial for classifiers to output confidence that reflects the ground truth corresponding to each class for each instance. If an ML model predicts with 50% confidence on a subset of examples, then ideally, it should classify at least 50% of the examples correctly. ML models exhibiting such behavior are called calibrated models. Methods such as Maximum Likelihood temperature scaling, MAP temperature scaling, Bayesian Scaling can be used.

3. **Regression Problem :** In our approach, we have trained the defer model by treating it as 0-1 classification problem. But we can also map this as a Regression problem, wherein the optimal target value is somewhere between 0 and 1. This might result in a better trained defer model.

4. **Multiple Humans :** We have only considered a single human expert. However, in real life we may have access to multiple as well. In that case, we need to identify the subset of humans to query in-order to reduce cost. This idea can also be taken forward with this approach.

References

[1] C. Chow. "On optimum recognition error and reject tradeoff". In: *IEEE Transactions on Information Theory* 16.1 (1970), pp. 41–46. DOI: 10.1109/TIT.1970.1054406.

[2] Gavin Kerrigan, Padhraic Smyth, and Mark Steyvers. "Combining Human Predictions with Model Probabilities via Confusion Matrices and Calibration". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 4421–4434. DOI: https://doi.org/10.48550/arXiv.2109.14591.

[3] David Madras, Toni Pitassi, and Richard Zemel. "Predict responsibly: Improving fairness and accuracy by learning to defer". In: *Advances in Neural Information Processing Systems* 31 (2018), pp. 6150–6160. DOI: https://doi.org/10.48550/arXiv.1711.06664.

[4] Hussein Mozannar and David Sontag. "Consistent Estimators for Learning to Defer to an Expert". In: *Proceedings of the 37th International Conference on Machine Learning* 119 (2020), pp. 7076–7087. DOI: https://doi.org/10.48550/arXiv.2006.01862.