Stop Overcomplicating Selective Classification: Use Max-Logit

Leo Feng
Mila & Borealis AI
Mohamed Osama Ahmed
Borealis AI
Hossein Hajimirsadeghi
Borealis AI
Amir Abdi
Borealis AI

Abstract

We tackle the problem of Selective Classification where the goal is to achieve the best performance on the desired coverages of the dataset. Recent state-of-the-art selective methods come with architectural changes either via introducing a separate selection head or an extra abstention logit. In this paper, we present surprising results for Selective Classification by confirming that the superior performance of state-of-the-art methods is owed to training a more generalizable classifier; however, their selection mechanism is suboptimal. We argue that the selection mechanism should be rooted in the objective function instead of a separately calculated score. Accordingly, in this paper, we motivate an alternative selection strategy that is based on the cross entropy loss for the classification settings, namely, max of the logits. Our proposed selection strategy achieves better results by a significant margin, consistently, across all coverages and all datasets, without any additional computation. Finally, inspired by our superior selection mechanism, we propose to further regularize the objective function with entropy-minimization. Our proposed max-logit selection with the modified loss function achieves new state-of-the-art results for Selective Classification.

1 Introduction

A model’s ability to abstain from a decision when lacking confidence is essential in mission-critical applications. This is known as the Selective Prediction problem setting. The abstained and uncertain samples can be flagged and passed to a human expert for manual assessment, which, in turn, can improve the re-training process. This is crucial in problem settings where confidence is critical or an incorrect prediction can have significant consequences such as in the financial, medical, or autonomous driving domains. Several papers have tried to address this problem by estimating the uncertainty in the prediction. Gal and Ghahramani [7] proposed using MC-dropout, Lakshminarayanan et al. [17] proposed to use an ensemble of models, while methods such as Dusenberry et al. [5] and Maddox et al. [21] are examples of work using Bayesian deep learning to estimate a probability distribution of the output. These methods are either expensive to train or require lots of tuning for acceptable results.

In this paper, we focus on the Selective Classification problem setting where a classifier has the option to abstain from making predictions. Models that come with an abstention option and tackle the selective prediction problem setting are naturally called selective models. Different selection approaches have been suggested such as incorporating a selection head [9] or an abstention logit [13, 29]. In either case, a threshold is set such that selection and abstention values above or below the threshold decide the selection action. SelectiveNet [9] proposes to learn a model comprising of a selection head and a prediction head where the values returned by the selection head determines whether the datapoint is selected for prediction or not. Huang et al. [13] and Ziyin et al. [29] introduced an additional abstention logit for classification settings where the output of the additional logit determines whether the model abstains from making predictions on the sample. The promising
results of these works suggest that the selection mechanism should focus on the output of the selection head or the abstain logit. On the contrary, in this work, we argue that the selection mechanism should be rooted in the objective function of classification; i.e., datapoints that lead to high prediction accuracy or low loss are best chosen for selection. Obviously, at test time, we do not have access to the true label so we can not calculate neither the accuracy nor the loss; yet, in the classification settings, there are a couple of alternatives. We empirically demonstrate that a selection mechanism based on the prediction class logits outperforms all the previously proposed selection methods.

Our contributions are as follows:

1. Propose a Selective Classification approach that uses the maximum predictive class logit without introducing any architectural changes, and empirically show that our proposal outperforms the three state-of-the-art Selective Classification methods (i.e., Self-Adaptive Training [13], Deep Gamblers [29], and SelectiveNet [9]).

2. Inspired by the strong empirical results of our selection mechanism, we take things one step further and propose an entropy-regularized loss function which outperforms the state-of-the-art Selective Classification methods by a significant margin (up to 60% relative improvement).

3. Going beyond the already-saturated datasets often used for Selective Classification research, we also include 2 larger datasets: Imagenet100 to test the methods on a wide range of coverages and ImagenetSubset to test the scalability of the methods.

2 Related Work

The option to reject a prediction has been explored in depth in various learning algorithms not limited to neural networks. Primarily, Chow [2] introduced a cost-based rejection model and analysed the error-reject trade-off. There has been significant study in rejection in Support Vector Machines [1, 6, 26, 27]. The same is true for nearest neighbours [12] and boosting [3] literature.

In 1989, LeCun et al. [18] proposed a rejection strategy for neural networks based on the most activated output logit, second most activated output logit, and the difference between the activated output logits. Geifman and El-Yaniv [8] presented a technique to achieve a target risk with a certain probability for a given confidence-rate function. As examples of confidence-rate functions, the authors suggested selecting according to the max predictive class logit, referring to it as Softmax Response, and MC-Dropout as selection mechanisms for a vanilla classifier. We build on this idea to demonstrate that Softmax Response, if utilized correctly, is the highest performing selection mechanism in the classification settings.

Geifman and El-Yaniv [9] proposed SelectiveNet (see Section 3.2.1), a three-headed model, comprising of heads for selection, prediction, and auxiliary prediction. Deep Gamblers [29] (see Section A.1) and Self-Adaptive Training [13] (see Section 3.3.1) proposes a \((C + 1)\)-way classifier, where \(C\) is the number of classes and the additional logit represents abstention. Ziyin et al. [29] compared selecting according to the model’s uncertainty, in particular, the prediction entropy, for a vanilla classifier as a baseline, but reported inferior results than Deep Gamblers. In contrast, in this work, we explain how selecting via entropy and max-logit can work as a proxy to select samples which could potentially minimise the cross entropy loss. In general, we report the surprising results that the selection head of the SelectiveNet and the abstention logits in Deep Gamblers and Self-Adaptive Training are suboptimal selection mechanisms and their previously reported good performance is rooted in their optimization process converging to a more generalizable model.

Finally, entropy minimization objectives have been widely used for Unsupervised Learning [20], Semi-Supervised Learning [10], and Domain Adaptation [25, 28]. In this work, we explore the entropy-regularized loss function in the Selective Classification settings to further improve the selective performance, which, to the best of our knowledge, has not been explored before.

3 Background

In this section, we introduce the Selective Classification problem. Additionally, we describe the top three methods for Selective Classification: SelectiveNet [9], Self-Adaptive Training [13], and Deep
Gambles [29]. To the best of our knowledge, Self-Adaptive Training achieves the best performance on the selective classification datasets.

3.1 Problem Setting: Selective Classification

The selective prediction task can be formulated as follows. Let \( \mathcal{X} \) be the feature space, \( \mathcal{Y} \) be the label space, and \( P(\mathcal{X}, \mathcal{Y}) \) represent the data distribution over \( \mathcal{X} \times \mathcal{Y} \). A selective model comprises of a prediction function \( f: \mathcal{X} \rightarrow \mathcal{Y} \) and a selection function \( g: \mathcal{X} \rightarrow \{0, 1\} \). The selective model decides to make predictions when \( g(x) = 1 \) and abstains from making predictions when \( g(x) = 0 \). The objective is to maximise the model’s predictive performance for a given target coverage \( c_{\text{target}} \in [0, 1] \), where coverage is the proportion of the selected samples. The selected set is defined as \( \{x : g(x) = 1\} \). Formally, an optimal selective model, parameterised by \( \theta^* \) and \( \psi^* \), would be the following:

\[
\theta^*, \psi^* = \arg \min_{\theta, \psi} \mathbb{E}_P[l(f_\theta(x), y) \cdot g_\psi(x)], \quad \text{s.t.} \quad \mathbb{E}_P[g_\psi(x)] \geq c_{\text{target}},
\]

where \( \mathbb{E}_P[l(f_\theta(x), y) \cdot g_\psi(x)] \) is the selective risk. Naturally, higher coverages are correlated with higher selective risks.

In practice, instead of a hard selection function \( g_\psi(x) \), existing methods aim to learn a soft selection function \( \bar{g}_\psi: \mathcal{X} \rightarrow \mathbb{R} \) such that larger values of \( \bar{g}_\psi(x) \) indicate the datapoint should be selected for prediction. At test time, a threshold \( \tau \) is selected for a coverage \( c \) such that

\[
g_\psi(x) = \begin{cases} 
1 & \text{if } \bar{g}_\psi(x) \geq \tau, \\
0 & \text{otherwise}
\end{cases}, \quad \text{s.t.} \quad \mathbb{E}[g_\psi(x)] = c_{\text{target}}
\]

In this setting, the selected (covered) dataset is defined as \( \{x : \bar{g}_\psi(x) \geq \tau\} \). The process of selecting the threshold \( \tau \) is known as calibration.

3.2 Approach: Learn to Select

3.2.1 SelectiveNet

SelectiveNet [9] is a three-headed network proposed for selective learning. A SelectiveNet model has three output heads designed for selection \( \bar{g} \), prediction \( f \), and auxiliary prediction \( h \). The selection head infers the selective score of each sample, as a value between 0 to 1, and is implemented with a sigmoid activation function. The auxiliary prediction head is trained with a standard (non-selective) loss function. The prediction head is trained to achieve the desired coverage and ensures \( \bar{g}(x_i) > 0 \) for at least \( c_{\text{target}} \) proportion of the batch samples. The selective loss \( L_{\text{selective}} \) discounts the weight of difficult samples via the soft selection value \( \bar{g}(x) \) term encouraging the model to focus more on easier samples which the model is more confident about.

The auxiliary loss \( L_{\text{aux}} \) ensures that all samples, regardless of their selective score \( \bar{g}(x) \), contribute to the learning of the feature model. \( \lambda \) and \( \alpha \) are hyper-parameters controlling the trade-off of different terms. Unlike Deep Gambles and Self-Adaptive Training, SelectiveNet trains a separate model for each target coverage \( c_{\text{target}} \). In the SelectiveNet paper [9], it has been suggested that the best performance is achieved when the training target coverage is equal to that of the evaluation coverage.

3.3 Approach: Learn to Abstain

Self-Adaptive Training [13] and Deep Gambles [29] propose to tackle the selective classification problem by introducing a \((C + 1)\)-th class logit where the extra class logit represents abstention. Let

\[
L_{\text{aux}} = \frac{1}{m} \sum_{i=1}^{m} \ell(h(x_i), y_i),
\]

where \( \ell \) is any standard loss function. In Selective Classification, \( \ell \) is the Cross Entropy loss function. The coverage loss \( L_c \) encourages the model to achieve the desired coverage and ensures \( \bar{g}(x_i) > 0 \) for at least \( c_{\text{target}} \) proportion of the batch samples. The selective loss \( L_{\text{selective}} \) discounts the weight of difficult samples via the soft selection value \( \bar{g}(x) \) term encouraging the model to focus more on easier samples which the model is more confident about.

The auxiliary loss \( L_{\text{aux}} \) ensures that all samples, regardless of their selective score \( \bar{g}(x) \), contribute to the learning of the feature model. \( \lambda \) and \( \alpha \) are hyper-parameters controlling the trade-off of different terms. Unlike Deep Gambles and Self-Adaptive Training, SelectiveNet trains a separate model for each target coverage \( c_{\text{target}} \). In the SelectiveNet paper [9], it has been suggested that the best performance is achieved when the training target coverage is equal to that of the evaluation coverage.
We motivate an alternative selection mechanism based on the maximum predictive class logit (\( \theta \)) where \( m \) is the label. Initially, the model is trained with a cross-entropy loss for a series of pre-training steps. Afterwards, the model is updated according to a dynamically moving training target. The training target \( t_i \) is initially set equal to the label \( t_i \leftarrow y_i \) such that the training target is updated according to \( t_i \leftarrow \alpha \times t_i + (1 - \alpha) \times p_\theta(\cdot | x_i) \) s.t. \( \alpha \in (0, 1) \) after each model update. Similar to Deep Gamblers, the model is trained to optimise a loss function that allows the model to also choose to abstain on hard samples instead of making a prediction:

\[
\mathcal{L} = -\frac{1}{m} \sum_{i=1}^{m} [t_i, y_i, \log p_\theta(y_i | x_i) + (1 - t_i, y_i) \log p_\theta(C + 1 | x_i)],
\]

where \( m \) is the number of datapoints in the batch. As training progresses, \( t_i \) approaches \( p_\theta(\cdot | x_i) \). The first term is similar to the Cross Entropy Loss and encourages the model to learn a good classifier. The second term encourages the model to abstain from making predictions for samples that the model is uncertain about. This use of dynamically moving training target \( t_i \) allows the model to avoid fitting on difficult samples as the training progresses.

4 Methodology

We motivate an alternative selection mechanism based on the maximum predictive class logit (i.e., Softmax Response) for a \( C \)-way classifier trained with the cross entropy loss. We explain how the known state-of-the-art selective methods can be equipped with the proposed selection mechanism and why it further improves performance. Accordingly, in Section 5 we empirically support our new top performer by reporting on its superiority on all the three explored datasets and across all coverages. Finally, inspired by the proposed selection mechanism, we introduce an entropy-regularized loss function and demonstrate that a model trained with the proposed objective function further improves the state-of-the-art results by a statistically significant margin.

4.1 Motivation

In the selective classification problem setting the objective is to select \( c_{\text{target}} \) proportion of samples for prediction according to the value outputted by a selection function, \( \tilde{g}(x) \). Since each datapoint \((x_i, y_i)\) is an i.i.d. sample, it is optimal to iteratively select from the dataset \( \mathcal{D} \) the sample \( x^* \) that maximizes the selection function, i.e., \( x^* \in \arg\max_{x \in \mathcal{D}} \tilde{g}(x) \)., until the target coverage \( c_{\text{target}} \) proportion of the dataset is reached. In other words, to select \( c_{\text{target}} \) proportion of samples (coverage = \( c_{\text{target}} \)), it is sufficient to define the criterion \( \tilde{g} \) and select a threshold \( \tau \) such that exactly \( c_{\text{target}} \) proportion of samples satisfy \( \tilde{g}(x) > \tau \).

4.1.1 Selecting via Maximum Predictive Class Logit (Softmax Response)

Let \( p_\theta(y | x) \) be a classifier parameterised by \( \theta \) trained on \( P(\mathcal{X}, \mathcal{Y}) \) with the cross entropy loss. As a result, \( p_\theta(y | x) \) can be interpreted as an approximation of the true distribution \( p(y | x) \). In supervised classification, the training objective is often the cross entropy loss, i.e., \( \text{CE}(p(\cdot | x), p_\theta(\cdot | x)) = - \sum_{y=1}^{C} p(u | x) \log p_\theta(u | x) \). The loss value of a sample is a metric for the model’s ability to predict the sample; hence, it would be most optimal to select samples with the lowest loss. However, during test time, labels are unknown. Instead, we can use the approximation \( p_\theta(y | x) \) instead of the true data distribution, resulting in the selection of samples that minimise the prediction entropy.
\[ CE(p_\theta(\cdot|x), p(x)) = H(p_\theta(\cdot|x)), \text{i.e., } \bar{g}(x) = -H(p_\theta(\cdot|x)) \]. In this setting, the entropy is also a notion of the model’s prediction uncertainty.

The cross entropy loss function is a popular loss function for classification due to its differentiability. However, during evaluation, the most utilized metric is accuracy, i.e., whether a datapoint is predicted correctly. In the cross-entropy objective of the conventional classification settings, \( p(c|x_i) \) is a one-hot encoded vector; therefore, the cross-entropy loss can be simplified as

\[ CE(p_\theta(\cdot|x_i), p_\theta(\cdot|x_i)) = -\sum_{u=1}^{C} p(u|x_i) \log p_\theta(u|x_i) = -\log p_\theta(y|x_i), \text{i.e., during optimization, the logit of the correct class is maximised.} \]

Accordingly, the maximum value of logits can be interpreted as the model’s relative confidence of its prediction. Therefore, the model can select according to the maximum predictive class logit, \( \bar{g}(x) = \max_{u \in \{1, \ldots, C\}} p_\theta(u|x_i) \) (aka Softmax Response \[8\]). An in-depth discussion with additional detail is included in the Appendix (see Section B).

### 4.2 Recipe for Better Selective Classification

Building on the motivations for the selection mechanism, we propose to perform selection according to the maximum predictive class logit (Softmax Response). So the recipe that we are providing for better selective classification is as follows:

1. Train a selective classifier (e.g., SelectiveNet, Self-Adaptive Training, or Deep Gamblers).
2. Discard its selection mechanism:
   - For SelectiveNet: Ignore the selection head
   - For Self-Adaptive Training and Deep Gambler: Ignore the additional abstain logit and compute the final layer’s softmax on the original \( C \) class logits.
3. Use the maximum predictive class logit (Softmax Response) to rank the samples.
4. Calculate the threshold value \( \tau \), based on the validation set, to achieve the desired target coverage and select samples with max logit greater than \( \tau \).

In Step 3, we also experimented with replacing the original selection mechanism with an entropy-based selection mechanism but we found that Softmax Response performed better. The detailed results are provided in [Section 5](#).

Notably, the proposed selection mechanism does not require any retraining of the model, hence, introducing no additional computational cost.

### 4.3 Entropy-regularized Loss Function

Taking into consideration the proposed selection mechanism, we propose to add an entropy minimization term to the objective function of the selection-abstention methods:

\[ L_{\text{new}} = L + \beta H(p_\theta(\cdot|x)), \]

where \( \beta \) is a hyperparameter that controls the impact. The entropy minimization term encourages the model to be more confident in its predictions, i.e., increasing the value of the maximum logit and decreasing the predictive entropy during training. The entropy minimization term discourages low-confidence predictions, allowing for better disambiguation between samples that should be selected and samples that should not. In [Section 5](#), we show that this proposed loss function in conjunction with our proposed selection mechanism further improves the selective performance.

### 5 Experiments

In this section, we introduce two new datasets for Selective Classification based on Imagenet [4, 23]: Imagenet100 [24] and ImagenetSubset. First, we show that our proposed selection mechanism, using max-logits on a pretrained selective classifier, outperforms the original selection mechanisms of the top three Selective Classification methods (SelectiveNet, Deep Gamblers, and Self-Adaptive Training) and achieves new state-of-the-art results on all the explored datasets. Furthermore, we show that even selecting according to the prediction entropy outperforms the mentioned methods which further supports our argument that selection should be rooted on the objective function. Finally, we demonstrate how our proposed entropy-regularized loss function applied on Self-Adaptive Training further improves upon the state-of-the-art results by a significant margin.
5.1 Datasets

In addition to CIFAR10, we introduce two datasets, Imagenet100 [24] and ImagenetSubset, for the selective classification problem setting and benchmark the existing state-of-the-art methods. We propose Imagenet100 as a realistic non-saturated dataset that can be evaluated at a wide range of coverages (10 – 100%). In addition, we propose ImagenetSubset as a collection of datasets to evaluate the scalability of the methods for different number of classes.

This is in contrast to the existing Selective Classification research which mainly have focused on small datasets with 10 or less classes, low resolution images (64x64 or less), and high coverages (70%+). The results of the previously introduced datasets indicate saturation, e.g., 99.7% accuracy at the 70% coverage, discouraging experiments with lower coverages, which, in turn, robs researchers of achieving conclusive results.

CIFAR-10. [16] The CIFAR-10 dataset comprises of small images: 50,000 images for training and 10,000 images for evaluation split into 10 classes. Each image is of size $32 \times 32 \times 3$. In our experiments, the selective classifiers are evaluated on coverages ranging from 70% to 100% in increments of 5%.

Imagenet100. We have chosen a subset of Imagenet [4, 23] originally proposed by Tian et al. [24], consisting of 100 classes sampled from the Imagenet dataset. The model is trained on the training dataset and evaluated on the validation dataset. The training data comprises of approximately 1,300 images per class. Evaluation is performed on the validation images, comprising of 5,000 images split into 100 classes. In our experiments, we evaluate classifiers on coverages between 10% and 100% in increments of 10%.

ImagenetSubset. We create a collection of datasets with varying number of classes (difficulty) from 25 to 175 in increments of 25. The classes are sampled randomly such that datasets with less classes are subsets of those with more classes. The complete list of selected classes in each dataset subset is available in the Appendix. These datasets help us evaluate the models’ performance with respect to the difficulty (scalability) of the task. Similar to Imagenet100, the training data comprises of approximately 1,300 images per class, and models are evaluated on the validation images, comprising of 50 images per class at a coverage of 70%.

5.2 Experiment Details

For our experiments, we used the publicly available official implementations of Deep Gamblers and Self-Adaptive Training. Experiments on SelectiveNet were conducted with our Pytorch implementation of the method which closely follows the details provided in the original paper [9].

For the CIFAR-10 experiment results to remain comparable, we follow the experimental details proposed in the original papers of SelectiveNet [9], Deep Gamblers [13], and Self-Adaptive Training [29].

For the Imagenet100 and ImagenetSubset datasets, we use a ResNet34 architecture for Deep Gamblers, Self-Adaptive Training, and the main body block of SelectiveNet. The models are trained for 500 epochs using ADAM [15] with a mini-batch size of 64. The learning rate was reduced by 0.5 every 25 epochs. We follow the original papers with regards to every other hyperparameter. Following [9], SelectiveNet was trained with a target coverage rate and evaluated on the same coverage rate. As a result, there are different models for each experimental coverage rate. In contrast, target coverage does not play a role in the optimization process of Deep Gamblers and Self-Adaptive Training, hence, the results for different experimental coverages are computed with the same models.

In the experiments involving the proposed entropy minimization loss function, the hyperparameter $\beta$ that controls the weight of the penalty is set to 0.01.

5.3 Results

For the following experiments, we evaluate the following state-of-the-art methods (1) SelectiveNet (SN), (2) Self-Adaptive Training (SAT), and (3) Deep Gamblers. Furthermore, we compare

\[\text{Note that the created dataset of ImagenetSubset with 100 classes is different than that of Imagenet100.}\]
Table 1: Comparison of SelectiveNet (SN) and Self-Adaptive Training (SAT) with Entropy (Ent.), Softmax Response (SR) as selection mechanisms on CIFAR-10. For a given selective classification method block (e.g., SelectiveNet), the first column refers to the original selection mechanism, the second column (Ent.) refers to selection via predictive entropy, and the third column (SR) refers to selection via Softmax Response. For a given selective classification method and coverage, the bolded result identifies the selection mechanism with the lowest (best) selective risk and underlined result identifies the selection mechanism with the second lowest (second best) selective risk.

| Cov. | SN  | SN + Ent. | SN + SR | SAT  | SAT + Ent. | SAT + SR |
|------|-----|-----------|---------|------|------------|----------|
| 100  | 6.47 ± 0.22 | 6.47 ± 0.22 | 6.47 ± 0.22 | 5.91 ± 0.04 | 5.91 ± 0.04 | 5.91 ± 0.04 |
| 95   | 4.07 ± 0.12  | 4.02 ± 0.11  | 4.05 ± 0.12  | 3.73 ± 0.11  | 3.59 ± 0.06  | 3.63 ± 0.10  |
| 90   | 2.49 ± 0.13  | 2.49 ± 0.13  | 2.49 ± 0.13  | 2.18 ± 0.11  | 2.12 ± 0.06  | 2.11 ± 0.06  |
| 85   | 1.42 ± 0.08  | 1.41 ± 0.08  | 1.43 ± 0.09  | 1.26 ± 0.09  | 1.18 ± 0.08  | 1.18 ± 0.07  |
| 80   | 0.86 ± 0.05  | 0.87 ± 0.05  | 0.86 ± 0.05  | 0.69 ± 0.04  | 0.63 ± 0.03  | 0.64 ± 0.04  |
| 75   | 0.53 ± 0.06  | 0.53 ± 0.05  | 0.54 ± 0.06  | 0.37 ± 0.01  | 0.35 ± 0.03  | 0.36 ± 0.03  |
| 70   | 0.42 ± 0.04  | 0.47 ± 0.06  | 0.47 ± 0.05  | 0.27 ± 0.02  | 0.23 ± 0.05  | 0.23 ± 0.05  |

The performance of these methods with various selection mechanisms (1) original selection mechanism, (2) Ent.:entropy, and (3) SR: Softmax Response (our proposed method).

The goal of our experimental evaluation is to answer the following questions: (1) What is the optimal selection mechanism for Selective Classification? (2) How scalable are the selective methods for larger datasets? (3) Does the proposed entropy-regularized loss function improve the model and the effectiveness of selection? Due to space limitations, we include the table results for Deep Gamblers and a vanilla classifier trained with cross-entropy in the Appendix.

5.3.1 CIFAR-10

We emphasize that we are particularly interested in the relative selective performance difference between the various selection methods.

Nonetheless, for this experiment, we attempt to replicate the previously published results for CIFAR-10. Based on our experiments, Self-Adaptive Training at coverages of 100, 90, 80, 75, and 70 performed slightly better than reported in the paper, however, it performed slightly worse than reported on coverages of 95 and 85 (Table 1). Similarly, SelectiveNet performed slightly better on coverages 100, 95, and 85 and slightly worse on coverages 90, 80, 75, and 70 than reported in the original paper (Table 1).

In Table 1, we are comparing the various selection mechanisms (i.e., original, Ent., and SR) on a given Selective Classification method (i.e., SelectiveNet and SAT). In our experiments, we found that Self-Adaptive Training with Softmax Response and prediction entropy as the selection mechanisms perform similarly, on average outperform both SelectiveNet and SAT, and achieve state-of-the-art results. Even though Softmax Response and prediction entropy demonstrated superior performance on average, the difference in selective performance amongst methods was not significant. We attribute the lack of a clear significant difference in performance amongst methods to the saturatedness of the small CIFAR-10 dataset. This is supported by the results in the next subsections.

5.3.2 Imagenet100

In Table 2, we compare the results for different selective classification models with different selection mechanisms. We see that the using Softmax Response as the selection mechanism significantly outperforms the original selection mechanism, achieving state-of-the-art results. In addition, we see that using prediction entropy as the selection mechanism is inferior to using Softmax Response but still outperforms the original selection mechanism by a substantial margin. Interestingly, at low coverages (30%, 20%, and 10%), SelectiveNet’s performance progressively gets worse. We hypothesize that this is due to the optimisation process of SelectiveNet that allows the model to disregard (i.e., assign lower weight to their loss) a vast majority of samples during training at little cost, i.e., \( \bar{g}(x) \approx 0 \), especially when the target coverage is as low as 10%. In contrast, Deep Gamblers and Self-Adaptive Training models are equally optimised over all samples regardless of their selection.
Table 2: SelectiveNet and Self-Adaptive-Training results on Imagenet100. Similar notations as Table 1.

Due to the space limitation, the results for Deep Gamblers and vanilla classifier are included in the Appendix. Briefly summarised, the additional interesting results found are as follows: (1) In low coverage settings, selecting based on Softmax Response and Entropy on a vanilla classifier trained via the cross entropy loss outperform both SelectiveNet and Deep Gamblers. (2) SelectiveNet outperforms Deep Gamblers on moderate coverages (60%, 50%, and 40%) which is not in par with the previously reported results. We attribute the interesting results to our work being the first to evaluate these methods on large datasets at a wide range of coverages. Since previous works have mainly focused on toy datasets and high coverages (70% +), they failed to capture these patterns. The main takeaway of these results, however, is that, across all the reported methods, selecting via Softmax Response is best.

### 5.3.3 ImagenetSubset

In this experiment (see Figure 1), we evaluate SelectiveNet and Self-Adaptive Training models on 70% coverage. Due to the space limitations, the table for the Deep Gamblers results are included in the Appendix. Consistent with the Imagenet100 experiments, we see that selecting according to Softmax Response is best.

Table 3: Results of SelectiveNet and Self-Adaptive Training with selection according to the predictive entropy (Ent.) and Softmax Response (SR) selection mechanism on ImagenetSubset.
**Table 4**: Results of Self-Adaptive Training with the proposed entropy minimization loss function (EM) on Imagenet100. Here, E and SR refer to selecting according to prediction entropy and Softmax Response, respectively.

| Cov. | SAT     | SAT + EM  | SAT + Entropy | SAT + EM + E | SAT + SR  | SAT + EM + SR |
|------|---------|-----------|---------------|--------------|-----------|---------------|
| 100  | 13.58 ± 0.30 | 13.18 ± 0.24 | 13.58 ± 0.30  | 13.18 ± 0.24  | 13.58 ± 0.30  | 13.18 ± 0.24  |
| 90   | 8.80 ± 0.41  | 8.69 ± 0.32  | 8.92 ± 0.43   | 8.72 ± 0.27   | 8.04 ± 0.25   | 7.73 ± 0.22   |
| 80   | 5.20 ± 0.29   | 5.03 ± 0.36   | 4.86 ± 0.22   | 4.65 ± 0.37   | 4.46 ± 0.13   | 3.90 ± 0.34   |
| 70   | 2.71 ± 0.29   | 2.61 ± 0.22   | 2.52 ± 0.23   | 2.35 ± 0.29   | 2.33 ± 0.19   | 1.81 ± 0.27   |
| 60   | 1.72 ± 0.11   | 1.59 ± 0.19   | 1.54 ± 0.15   | 1.44 ± 0.15   | 1.37 ± 0.12   | 0.95 ± 0.13   |
| 50   | 1.18 ± 0.14   | 1.02 ± 0.21   | 1.03 ± 0.12   | 0.94 ± 0.19   | 0.88 ± 0.07   | 0.62 ± 0.09   |
| 40   | 0.82 ± 0.06   | 0.81 ± 0.12   | 0.77 ± 0.07   | 0.78 ± 0.07   | 0.60 ± 0.11   | 0.34 ± 0.06   |
| 30   | 0.67 ± 0.06   | 0.61 ± 0.14   | 0.56 ± 0.08   | 0.60 ± 0.12   | 0.59 ± 0.11   | 0.25 ± 0.10   |
| 20   | 0.48 ± 0.18   | 0.52 ± 0.16   | 0.40 ± 0.18   | 0.48 ± 0.13   | 0.46 ± 0.22   | 0.15 ± 0.08   |
| 10   | 0.32 ± 0.10   | 0.32 ± 0.20   | 0.32 ± 0.10   | 0.32 ± 0.20   | 0.12 ± 0.16   | 0.12 ± 0.07   |

max logit (Softmax Response) improves upon the originally proposed selection mechanisms by a significant margin. In addition, even selecting according to the prediction entropy outperforms the original selection mechanism. In Table 3, we see that Self-Adaptive Training with Softmax Response as the selection mechanism consistently performs best across all the various dataset sizes.

**Scalability.** In the experiments, we see that SelectiveNet struggles to scale to harder tasks (Table 3). Accordingly, the achieved improvement in selective accuracy with Softmax Response (SR) increases as the number of classes increase. This suggests that the proposed selection mechanism is more beneficial for SelectiveNet as the difficulty of the task increases, i.e., improves scalability. Interestingly, we see that the beneficial effect of selecting via Softmax Response or entropy decreases for Deep Gamblers and Self-Adaptive Training as the number of classes increase.

### 5.3.4 Entropy-regularized Loss Function

In this experiment, we evaluate the efficacy of the proposed loss function (Section 4.3) in training a selective classifier. In this experiment, we focus on Self-Adaptive Training since it was the best performing method. As demonstrated in Table 4, training with the entropy minimization loss function converges to a more optimal model. More importantly, a clear and considerable improvement can be observed across all coverages with Softmax Response as the selection mechanism for Self-Adaptive Training trained with the proposed entropy-regularized loss function, e.g., 62.5%, 68.8%, 62.7%, and 58.5% relative improvement at 10%, 20%, 30%, and 40% coverages, respectively.

### 6 Conclusion

In this work, we analysed the state-of-the-art Selective Classification methods and concluded that although these methods learn a more generalizable model, their selection method is not the best. We motivate an alternative selection mechanism based on max class logit, which is applicable to both vanilla classifiers and other selective models such as SelectiveNet, Deep Gamblers, and Self-Adaptive Training. We empirically show that selecting via max class logit achieves state-of-the-art results across all datasets without additional compute cost. Furthermore, we suggest entropy-regularized loss function to further improve upon the state-of-the-art by a considerable margin. In contrast with previous work, we assess all the models on low coverages and show that SelectiveNet struggles in low coverages and fails to scale to difficult problems where the level of difficulty is defined by the number of classes. With the rigorously conducted experiments, we advise any future research on selection strategies to consider the max of the logits of the predictive model as the primary state-of-the-art method for comparison.

**Broader impact.** The broader impact of this work depends on the application of the selective model. In terms of the societal impact, fairness in selection remains a concern as lowering the coverage can magnify the difference in recall between groups and increase unfairness [14][19].
References

[1] P. L. Bartlett and M. H. Wegkamp. Classification with a reject option using a hinge loss. *J. Mach. Learn. Res.*, 9:1823–1840, jun 2008. ISSN 1532-4435.

[2] C. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970. doi: 10.1109/TIT.1970.1054406.

[3] C. Cortes, G. DeSalvo, and M. Mohri. Boosting with abstention. *Advances in Neural Information Processing Systems*, 29, 2016.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, pages 248–255. Ieee, 2009.

[5] M. Dusenberry, G. Jerfel, Y. Wen, Y. Ma, J. Snoek, K. Heller, B. Lakshminarayanan, and D. Tran. Efficient and scalable Bayesian neural nets with rank-1 factors. In *International Conference on Machine Learning*, 2020.

[6] G. Fumera and F. Roli. Support vector machines with embedded reject option. In *International Workshop on Support Vector Machines*, pages 68–82. Springer, 2002.

[7] Y. Gal and Z. Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, 2016.

[8] Y. Geifman and R. El-Yaniv. Selective classification for deep neural networks. *Advances in neural information processing systems*, 30, 2017.

[9] Y. Geifman and R. El-Yaniv. Selectivenet: A deep neural network with an integrated reject option. In *International Conference on Machine Learning*, pages 2151–2159. PMLR, 2019.

[10] Y. Grandvalet and Y. Bengio. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004.

[11] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2017.

[12] M. E. Hellman. The nearest neighbor classification rule with a reject option. *IEEE Trans. Syst. Sci. Cybern.*, 6:179–185, 1970.

[13] L. Huang, C. Zhang, and H. Zhang. Self-adaptive training: beyond empirical risk minimization. *Advances in neural information processing systems*, 33:19365–19376, 2020.

[14] E. Jones, S. Sagawa, P. W. Koh, A. Kumar, and P. Liang. Selective classification can magnify disparities across groups. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=N0M_4BkQ05i.

[15] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[16] A. Krizhevsky. Learning multiple layers of features from tiny images. 2009.

[17] B. Lakshminarayanan, A. Pritzel, and C. Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, 2017.

[18] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel. Handwritten digit recognition with a back-propagation network. *Advances in neural information processing systems*, 2, 1989.

[19] J. K. Lee, Y. Bu, D. Rajan, P. Sattigeri, R. Panda, S. Das, and G. W. Wornell. Fair selective classification via sufficiency. In M. Meila and T. Zhang, editors, *International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 6076–6086. PMLR, 18–24 Jul 2021.

[20] M. Long, H. Zhu, J. Wang, and M. I. Jordan. Unsupervised domain adaptation with residual transfer networks. *Advances in neural information processing systems*, 29, 2016.

[21] W. J. Maddox, P. Izmailov, T. Garipov, D. P. Vetrov, and A. G. Wilson. A simple baseline for bayesian uncertainty in deep learning. *Advances in Neural Information Processing Systems*, 2019.

[22] M. Minderer, J. Djolonga, R. Romijnders, F. Hubis, X. Zhai, N. Houlsby, D. Tran, and M. Lučić. Revisiting the calibration of modern neural networks. *Advances in Neural Information Processing Systems*, 34, 2021.
[23] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[24] Y. Tian, D. Krishnan, and P. Isola. Contrastive multiview coding. In *European conference on computer vision*, pages 776–794. Springer, 2020.

[25] T.-H. Vu, H. Jain, M. Bucher, M. Cord, and P. Pérez. Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2517–2526, 2019.

[26] M. Wegkamp. Lasso type classifiers with a reject option. *Electronic Journal of Statistics*, 1:155–168, 2007.

[27] M. Wegkamp and M. Yuan. Support vector machines with a reject option. *Bernoulli*, 17(4):1368–1385, 2011.

[28] X. Wu, Q. Zhou, Z. Yang, C. Zhao, L. J. Latecki, et al. Entropy minimization vs. diversity maximization for domain adaptation. *arXiv preprint arXiv:2002.01690*, 2020.

[29] L. Ziyin, Z. T. Wang, P. P. Liang, R. Salakhutdinov, L.-P. Morency, and M. Ueda. Deep gamblers: learning to abstain with portfolio theory. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pages 10623–10633, 2019.
A Appendix: Additional Background

A.1 Deep Gamblers

Inspired by portfolio theory, Deep Gamblers proposes to train the model using the following loss function:

$$\mathcal{L} = -\frac{1}{m} \sum_{i=1}^{m} p(y_i|x) \log \left( p_{\theta}(y_i|x) + \frac{1}{o} p_{\theta}(C + 1|x) \right),$$

where $m$ is the number of datapoints in the batch and $o$ is a hyperparameter controlling the impact of the abstain logit. Smaller values of $o$ encourages the model to abstain more often. However, $o \leq 1$ makes it ideal to abstain for all datapoints and $o > C$ makes it ideal to predict for all datapoints. As a result, $o$ is restricted to be between 1 and $C$. Note that the corresponding loss function with large values of $o$ is approximately equivalent to the Cross Entropy loss.

B Appendix: Motivation

B.1 Selecting via Predictive Entropy

At test time, given a dataset of datapoints $D$, if the labels were available, the optimal criterion to select the datapoint $x \in D$ that minimise the loss function would be according to:

$$\arg\min_{x \in D} CE(p(\cdot|x), p_{\theta}(\cdot|x)).$$

However, at test time, the labels are unavailable. Instead, we can use the model’s belief over what the label is, i.e., the learned approximation $p_{\theta}(\cdot|x) \approx p(\cdot|x)$. We know $CE(p_{\theta}(\cdot|x), p_{\theta}(\cdot|x)) = H(p_{\theta}(\cdot|x))$ where $H$ is the entropy function. As such, we can select samples according to

$$\arg\min_{x \in D} CE(p(\cdot|x), p_{\theta}(\cdot|x)) \approx \arg\min_{x \in D} H(p_{\theta}(\cdot|x)).$$

In other words, entropy is an approximation for the unknown loss function. Accordingly, with respect to the discussed selection framework (Section 3.1), the samples with the largest negative entropy value, i.e., $\bar{g}(x) = -H(p_{\theta}(\cdot|x))$ are best nominees for selection.

Figure 2: A histogram of the number of datapoints according to a vanilla classifier trained on Imagenet100. The orange bar indicates the samples for which the model correctly predicts the class of the sample. The blue bar represents the samples for which the model incorrectly predicted the class. In the case of entropy, a lower value corresponds to higher model confidence. In contrast, in the case of max class logit, a higher value corresponds to higher model confidence. In Figure 2 we show the distribution of entropy for a trained vanilla classifier, empirically showing entropy to be strongly inversely correlated with the model’s ability to correctly predict the labels. As a result, entropy is a good selection mechanism. We include results on CIFAR-10 and Imagenet100 for a vanilla classifier in Table 5 and Table 6.

B.2 Selecting via Maximum Predictive Class Logit (Softmax Response)

Given a model with well-calibrated confidences [11, 22], an interpretation of $p_{\theta}(u|x)$ is a probability estimate of the true correctness likelihood, i.e., $p_{\theta}(u|x)$ is the likelihood that $u$ is the correct
Figure 3: Entropy Comparison. SelectiveNet trained on Imagenet100 for a target coverage of 0.8 and evaluated on a coverage of 0.8. In the case of entropy, a lower value corresponds to higher model confidence. The histogram represents the counts of samples that were incorrectly predicted by the model. The left image indicates datapoints that were not selected by the selection head, i.e., datapoints with low selection value $h(x) < \tau$. The right image indicates datapoints that were selected by the selection head, i.e., $h(x) \geq \tau$.

Figure 4: Max Class Logit Comparison. SelectiveNet trained on Imagenet100 for a target coverage of 0.8 and evaluated on a coverage of 0.8. In the case of max class logit, a higher value corresponds to higher model confidence. The histogram represents the counts of samples that were incorrectly predicted by the model. The left image indicates datapoints that were not selected by the selection head, i.e., datapoints with low selection value $h(x) < \tau$. The right image indicates datapoints that were selected by the selection head, i.e., $h(x) \geq \tau$.

Label of $x_i$. Let $y_i$ be the correct label for $x_i$. For example, given 100 samples $\{x_1, \ldots, x_{100}\}$ with $p_\theta(y_i|x_i) = 0.8$, we would expect approximately 80% of the samples to have $u$ as its label. As a result, $p_\theta(y_i|x_i)$ is the model’s probability estimate that the correct label is $y_i$. In classification, the probability that the calibrated model predicts a datapoint $x$ correctly is equivalent to the value of the max class logit, i.e., $\max_{u \in \{1, \ldots, C\}} p_\theta(u|x)$. Logically, the sample $x_i$ that should be selected for classification is the sample the model’s most likely to predict the sample correctly, i.e., $i = \arg\max_j \left( \max_{u \in \{1, \ldots, C\}} p_\theta(u|x_j) \right)$. This selection is equivalent to selecting according to the following soft selection function $\bar{g}(x) = \max_{u \in \{1, \ldots, C\}} p_\theta(u|x)$. Simply put, this is equivalent to selecting according to the maximum predictive class logit (aka Softmax Response [8]).

In practice, neural network models are not guaranteed to have well-calibrated confidences. In Selective Classification, however, we threshold according to $\tau$ and select samples above the threshold $\tau$ for classification, so we do not use the exact values of the confidence (max class logit). As a result, we do not need the model to necessarily have well-calibrated confidences. Instead, it suffices if samples with higher confidences (max class logit) have a higher likelihood of being correct. In Figure 2b, we show the distribution of max class logit for a trained vanilla classifier, empirically showing larger max class logit to be strongly correlated with model’s ability to correctly predict the label. As a result, max class logit is a good selection mechanism. We include results on CIFAR-10 and Imagenet100 for a vanilla classifier in Table 5 and Table 6.
B.3 Recipe for Better Selective Classification

In this section, we further illustrate how SelectiveNet’s original selection mechanism is suboptimal. The optimisation of SelectiveNet’s selective loss $L_{\text{selective}}$ (See Section 3.2.1) aims to learn a selection head (soft selection model) $\tilde{g}$ that outputs a low selection value for inputs with large cross-entropy loss and high selection value for inputs with low cross-entropy loss. At test time, good performance of SelectiveNet depends on the generalisation of both the prediction and selection heads. However, learned models can at times fail to generalise. In Figure 3 and Figure 4, we show the distribution of entropy and max class logit for selected and not-selected samples according to a SelectiveNet model. In the plots, we see SelectiveNet’s original selection mechanism selects several samples with large entropy and low max class logit. In Table 2, we see that the selection mechanisms based on entropy and max class logit outperforms the original selection mechanism. This comparison further supports our argument that the selection mechanism should be rooted in the objective function instead of a separately calculated score.

C Appendix: Additional Experimental Details

C.1 Hyperparameters

All CIFAR-10 experiments were performed with 5 seeds. All Imagenet-related experiments were performed with 3 seeds. For hyperparameter tuning, we split Imagenet100’s training data into 80% training data and 20% validation data evenly across the different classes. We tested the following values for the entropy minimization coefficient $\beta \in \{0.001, 0.01, 0.1\}$ and concluded that $\beta = 0.01$ performs best. For the final evaluation, we trained the model on the entire training data.

C.2 Compute

The experiments were primarily run on a GTX 1080 Ti. The CIFAR10 experiments took 1.5 hours for Self-Adaptive Training and Deep Gamblers. SelectiveNet experiments took 3 hours each. The Imagenet100 experiments took 2 days for Self-Adaptive Training and Deep Gamblers. SelectiveNet experiments took 2.75 days each. The ImagenetSubset experiments took 0.5-4.5 days each for Self-Adaptive Training and Deep Gamblers, depending on the number of classes. SelectiveNet experiments took 0.75-5.5 days each, depending on the number of classes.

C.3 ImagenetSubset: Classes

ImagenetSubset comprises of multiple datasets ranging from 25 to 175 classes in increments of 25, i.e., $\{D_{25}, D_{50}, D_{75}, D_{100}, D_{125}, D_{150}, D_{175}\}$. Let $C_{25}, C_{50}, \ldots, C_{175}$ represent the classes of the respective datasets. The classes for ImagenetSubset are uniform randomly sampled from the classes of Imagenet such that the classes of the smaller datasets are subsets of the classes of the larger datasets, i.e. $D_{25} \subset D_{50} \subset D_{75} \subset \cdots \subset D_{175}$ and $C_{25} \subset C_{50} \cdots \subset C_{175}$. The list of Imagenet classes in each dataset is included below for reproducibility.

C.3.1 $C_{25}$

n03133878 n03983396 n03995372 n03776460 n02730930 n03814639 n03666591 n03110669 n04442312 n02017213 n04265275 n01774750 n03709823 n09256479 n07715103 n04560804 n02120505 n04522168 n04074963 n02268443 n03291819 n02091467 n02486261 n03180011 n02100236

C.3.2 $C_{50} - C_{25}$

n02106662 n01871265 n12057211 n04579432 n07734744 n02408429 n02025239 n03649909 n03041632 n02484975 n02097209 n03854065 n03476684 n04579145 n01739381 n02319022 n01843383 n02229544 n09288635 n02138441 n02119022 n07583066 n03534580 n02817516 n04356056
C.3.3 $C_{75} - C_{50}$
n03424325 n04507155 n02112350 n03450230 n01616318 n01641577 n03630383 n01530575
n02102973 n04310018 n02134084 n01729322 n03250847 n02099849 n03544143 n03871628
n03777754 n04465501 n01770081 n03255030 n01910747 n03016953 n03485407 n03998194
n02129604

C.3.4 $C_{100} - C_{75}$
n02128757 n03763968 n01677366 n03483316 n02177972 n03814906 n01755581 n02264363
n03290653 n03814906 n01313613 n03929660 n04040759 n02317335 n02494079
n02865351 n03134739 n02102177 n04192698 n02814533 n04090263 n01818515 n01748264
n04328186

C.3.5 $C_{125} - C_{100}$
n03930313 n02422106 n07714571 n02111277 n03706229 n03729826 n03344393 n07831146
n02090379 n06596364 n03187595 n04317175 n11939491 n04277352 n01807496 n02804610
n02093991 n09428293 n03207941 n02132136 n04548280 n02793495 n03924679 n02112137
n02107312

C.3.6 $C_{150} - C_{125}$
n03376595 n03467068 n02837789 n04467665 n04243546 n03530642 n04398044 n02113624
n13044778 n03188531 n07714571 n01729977 n01980166 n02101388 n01629819 n01773157
n01689811 n02109525 n03938244 n02123045 n04548362 n04612504 n04264628 n02108551
n04311174 n02276258

C.3.7 $C_{175} - C_{150}$
n03724870 n02087046 n09421951 n02799071 n07717410 n02906734 n02206856 n03877472
n01740131 n04523525 n03496892 n04116512 n03743016 n03759954 n04462240 n03788195
n02137549 n03866082 n02233338 n02219486 n02445715 n02974003 n01924916 n12620546
n02992211

D Appendix: Additional Results

D.1 Vanilla Classifier

D.1.1 CIFAR-10

In Table 5, the difference in performance between selecting according to entropy and selecting
according to Softmax Response is not significant. We attribute this marginal difference to the
saturatedness of the CIFAR-10 dataset.

D.1.2 Imagenet100

In Table 6, we see that selecting according to Softmax Response clearly outperforms selecting
according to entropy. We see that Softmax Response learns a less generalizeable classifier (See
performance on 100% coverage) than Self-Adaptive Training, Deep Gamblers, and SelectiveNet.
However, interestingly, we found that Softmax Response outperforms both Deep Gamblers and
SelectiveNet on low coverages (10%, 20%, 30%). Previous works failed to capture this pattern due
to lack of evaluation on larger datasets and lower coverages.

D.2 Entropy-Minimization Results

D.2.1 ImagenetSubset

In addition to the Imagenet100 experiments, we also evaluate Self-Adaptive Training trained with the
proposed entropy-regularised loss function on ImagenetSubset. In Table 7, we see that Self-Adaptive
Table 5: Comparison of selection based on Entropy and Softmax Response for a vanilla classifier trained with cross-entropy loss on CIFAR-10. For a given selective coverage, the bolded result identifies the selection mechanism with the lowest (best) selective risk and underlined result identifies the selection mechanism with the second lowest (second best) selective risk.

Table 6: Comparison of selection based on Entropy and Softmax Response for a vanilla classifier trained with cross-entropy loss on Imagenet100. For a given selective coverage, the bolded result identifies the selection mechanism with the lowest (best) selective risk and underlined result identifies the selection mechanism with the second lowest (second best) selective risk.

Training with Entropy-Minimization further improves performance significantly on ImagenetSubset, achieving state-of-the-art results.

In Figure 5 (and Table 7), we see that training with the entropy-regularised loss function improves the scalability of Self-Adaptive Training when selecting according to Softmax Response.

Figure 5: Comparison of Self-Adaptive Training trained with the proposed entropy-regularised loss and without. The loss function improves the scalability of Self-Adaptive Training, particularly when using Softmax Response as the selection mechanism.
Table 7: Self Adaptive Training with Entropy Minimization Loss Function on Imagenet Subset. We see that SAT + EM + SR performs the best and outperforms SAT by a statistically significant margin.

D.3 Deep Gamblers Results

**CIFAR-10.** In these results (Table 8), we see that the difference in performance between the various selection mechanisms is marginal. Similar to the CIFAR-10 results for SelectiveNet and Self-Adaptive Training, it is difficult to make conclusions from these results.

**Imagenet100.** In Table 8, we see that selecting according to Softmax Response clearly outperforms the original selection mechanism. Furthermore, selecting according to entropy also outperforms the original selection mechanism.

**ImagenetSubset.** In Table 8, similar to Imagenet100, we see a clear substantial improvement when using Softmax Response as the selection mechanism instead of the original selection mechanism. Furthermore, we see that Entropy also outperforms the original selection mechanism.

| Dataset       | Coverage | Deep Gamblers          |         |         |         |
|---------------|----------|------------------------|---------|---------|---------|
|               |          | DG                     | DG + Entropy | DG + SR |
| CIFAR-10      | 100      | 6.08 ± 0.00            | 6.08 ± 0.00 | 6.08 ± 0.00 |
|               | 95       | 3.71 ± 0.00            | 3.79 ± 0.00 | 3.81 ± 0.00 |
|               | 90       | 2.27 ± 0.00            | 2.14 ± 0.00 | 2.16 ± 0.00 |
|               | 85       | 1.29 ± 0.00            | 1.31 ± 0.00 | 1.35 ± 0.00 |
|               | 80       | 0.81 ± 0.00            | 0.84 ± 0.00 | 0.85 ± 0.00 |
|               | 75       | 0.44 ± 0.00            | 0.57 ± 0.00 | 0.56 ± 0.00 |
|               | 70       | 0.30 ± 0.00            | 0.41 ± 0.00 | 0.43 ± 0.00 |
| Imagenet100   | 100      | 13.49 ± 0.52           | 13.49 ± 0.52 | 13.49 ± 0.52 |
|               | 90       | 8.42 ± 0.44            | 8.25 ± 0.43 | 8.11 ± 0.48 |
|               | 80       | 5.21 ± 0.32            | 4.76 ± 0.37 | 4.52 ± 0.38 |
|               | 70       | 3.30 ± 0.40            | 2.70 ± 0.21 | 2.58 ± 0.21 |
|               | 60       | 2.14 ± 0.37            | 1.86 ± 0.32 | 1.71 ± 0.32 |
|               | 50       | 1.55 ± 0.27            | 1.35 ± 0.25 | 1.31 ± 0.22 |
|               | 40       | 1.23 ± 0.38            | 1.20 ± 0.11 | 1.07 ± 0.19 |
|               | 30       | 1.09 ± 0.31            | 1.00 ± 0.19 | 0.96 ± 0.21 |
|               | 20       | 1.03 ± 0.31            | 0.97 ± 0.21 | 0.90 ± 0.22 |
|               | 10       | 0.80 ± 0.28            | 0.73 ± 0.25 | 0.53 ± 0.25 |

Table 8: Deep Gamblers Results on CIFAR-10 and Imagenet100. Comparison of Selection Mechanism Results.
## Deep Gamblers Results on ImagenetSubset with various selective mechanisms.

| Dataset  | # of Classes | Deep Gamblers | DG               | DG + Entropy | DG + SR            |
|----------|--------------|---------------|------------------|--------------|--------------------|
| Imagenet Subset | 175          |               | 3.77 ± 0.10      | 3.75 ± 0.14  | 3.62 ± 0.11        |
|          | 150          |               | 2.62 ± 0.03      | 2.65 ± 0.26  | 2.54 ± 0.24        |
|          | 125          |               | 2.58 ± 0.25      | 2.40 ± 0.19  | 2.22 ± 0.17        |
|          | 100          |               | 2.57 ± 0.04      | 2.30 ± 0.01  | 2.20 ± 0.04        |
|          | 75           |               | 2.60 ± 0.20      | 2.29 ± 0.00  | 2.22 ± 0.05        |
|          | 50           |               | 2.63 ± 0.12      | 2.15 ± 0.05  | 2.08 ± 0.07        |
|          | 25           |               | 1.60 ± 0.28      | 1.22 ± 0.30  | 1.30 ± 0.19        |

Table 9: Deep Gamblers Results on ImagenetSubset with various selective mechanisms.