

# Improvement of Rejection for AI Safety through Loss-Based Monitoring

Daniel Scholz<sup>1,3,\*</sup>, Florian Hauer<sup>2</sup>, Klaus Knobloch<sup>1</sup> and Christian Mayr<sup>3</sup>

<sup>1</sup>*Infineon Technologies Dresden, Königsbrücker Str. 180, 01099 Dresden, Germany*

<sup>2</sup>*Infineon Technologies München, Am Campeon 1-15, 85579 Neubiberg, Germany*

<sup>3</sup>*Technische Universität Dresden, Chair of Highly-Parallel VLSI Systems and Neuro-Microelectronics, Mommsenstr. 12, 01069 Dresden, Germany*

## Abstract

There are numerous promising applications for AI which are safety-critical, e.g. computer vision for automated driving. This requires safety measures for the underlying algorithm. Typically, the validity of a classification is solely based on the output probability of a network. Literature suggests that by rejecting classifications below an a-priori set probability threshold, the error rate of the network can be reduced. This inherently does not catch those errors, where the output probability of wrong classifications exceeds such a threshold. However, these are the most critical errors, since the system is erroneously overconfident. To solve this problem and close the gap, we present how this rejection idea can be improved by performing loss-based rejection. Our approach takes data as well as the pre-trained base-model as input and yields a monitoring model as output. For training of the monitoring model, the data samples are labeled based on the loss resulting from the base-model. This way, overconfident misclassifications can be avoided and the overall error rate reduced. As evaluation, we applied the approach to two datasets, one of which is the German Traffic Sign Recognition Benchmark (GTSRB) that is used to train safety-critical traffic sign classifiers. The experiments show that this approach yields results that improve the error-rate up to an order of magnitude while a portion of inputs is rejected as trade-off.

## Keywords

Rejection, AI Safety, Robustness, Classification, Neural Networks, Representation Learning,

## 1. Introduction

Artificial neural networks (ANN) are deployed for a variety of tasks. If the present trend persists, they will be more frequently included for safety critical decisions in fields such as medical diagnosis or automated driving. Therefore, safety of AI is important and already broadly discussed [1, 2, 3].

For safety-critical domains, e.g. the automotive domain, there exists no suitable standard for the safety assessment of ANNs yet [4, 5]. Future standardized safety assessments might aim for a high test set accuracy or ensure a low error-rate. Latter is especially important for safety critical applications. Error-rates directly result from the accuracy if and only if a prediction is forced in all cases and no reject option exists. For this work, a model is considered safer compared to another model for the same task if the relative error-rate is reduced using the same evaluation method in the same testing context.

Upon integration in a running system, possibly combined with multiple sensors and algorithms, it must be

judged whether single predictions of an ANN are trustworthy. This is mandatory to decide whether actions are performed based on those predictions or a verified safety path shall be used as fallback solution [6]. Especially, when the softmax activation function is applied at the output, resulting values are interpreted as probabilities and might be mistakenly used as a confident measure of the given prediction [7]. It has been shown that when those networks are trained with multinomial cross-entropy loss a tendency of over-confident decisions exists [8]. Usually the term “over-confident predictions” implies that the error-rate does not match the reported output probability for a given prediction. In the following “over-confident predictions” includes cases with relatively high output probabilities that might reflect the true error-rate but are incorrectly predicted. Such errors are the worst kind of failures that an ANN may produce. A deployed system might rely on the reported high confidence of the model and will not - without any additional mechanism - be able to maintain a safe state.

**Problem:** A decision mechanism is required to decide whether or not a prediction should be forced on an input, when it is necessary to reduce the error-rate beyond a model’s performance.

Different methods are present to approach those problems. Proper calibration [8, 9] enables to reduce the impact of over-confident errors. However, noting the extended definition of over-confident errors in this work

*The IJCAI-ECAI-22 Workshop on Artificial Intelligence Safety (AISafety 2022), July 24–25, 2022, Vienna, Austria*

\*Corresponding author.

✉ daniel.scholz@infineon.com (D. Scholz);


florian.hauer@infineon.com (F. Hauer);

klaus.knobloch@infineon.com (K. Knobloch);

christian.mayr@tu-dresden.de (C. Mayr)

© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License

Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

it does not address the actual issue. Other works [10, 11, 12, 13] address the issue directly by rejecting samples. Some use a selective prediction score that is built-in and improved during the initial training. In addition, there exist approaches that result in mathematical heuristics upon which the decisions to abstain are made. However, output probabilities for which rejections are expected are rarely reported. Decreasing the error-rate solely by discarding decisions which report a low output probability is less noteworthy for safety-critical domains since one would not rely on such decisions in the first place. **Summarized, there exists no approach that reduces the error rate of a present blackbox model by rejection based on a trained representation of the model’s weak points that can detect over-confident errors.**

In the following the model which is monitored is referred to as the “base-model” while the additional one is called the “monitoring model”. We close this gap with the following contribution: We present how the well-known rejection procedure can be improved by proposing a loss-based rejection. Our approach yields an monitoring model as output via training centered on the base-model’s loss. This way, overconfident misclassifications can be avoided and the overall error rate reduced.

## 2. Related Work

There are multiple approaches which can be considered to have the same goal of improving safety in AI. A trust score was proposed that is supposed to correlate with whether the classification is actually correct. It measures the consensus between the base classifier and a modified nearest-neighbor classifier during test case [14]. For the Digits dataset it was possible to detect trustworthy and suspicious predictions. The authors stated that for higher dimensional datasets like MNIST the trust score provides only little or no improvement in detecting wrong decisions better than the base model’s confidence itself.

Another line of research suggests to use the data’s distribution: Present approaches perform anomaly-detection on a dataset [15]. This flags specific samples but is purely based on the data and does not include any information about the model or the training process. More specifically, for a fixed dataset but models with different weak points, the same samples would be identified as possible failures, since the models are not part of the evaluation. Similar applies when out-of-distribution detection [16] is performed. It can be distinguished between data near and far away from the training distribution, typically corresponding to a different dataset. However, this does not prove an improvement for test data that lies inside the training distribution.

Bayesian neural networks (BNN) [17] or the applica-

tion of Monte Carlo Dropout [7] change the classical deterministic fashion of ANNs to a probabilistic nature, therefore, equipping the model itself with a built-in robust nature. Both methods increase generalization but do not necessarily perform better on crucial samples. Additionally, there is a computational overhead due to multiple network evaluations to calculate the output for a single input. Intuitively, these works are important, but simply address a different problem.

Approaches that include rejection [10, 18, 19, 11, 12, 13] show the same underlying principle as in this work. Rejection is commonly trained in combination with the classifier. However, it is advantageous if the monitoring approach that is used to abstain must not necessarily be trained combined with a base-model like shown in this work. Additionally, it is commonly focused on reduction of error-rate without differentiating between rejected predictions with a low and high output probability.

More straightforward methods like a probability or confidence score threshold under which a prediction is not trusted [18, 20, 21, 19] will be effective in decreasing the overall error rate. However, since such a threshold ideally divides confident and uncertain cases, per definition it will fail to catch over-confident predictions. Considering the above, there exists no model specific decision boundary to choose which prediction to trust that enables to exclude also over-confident decisions, leaving the risk of possible fatal situations for high output probabilities unchanged.

## 3. Preliminaries and Formalism

**Selective Prediction.** When rejection also known as selective prediction [22] is performed for a classification problem, it can be formulated as follows. Let  $\mathcal{X}$  be a feature space with its corresponding class label space  $\mathcal{Y}$  that enables supervised training. In this work  $\mathcal{X}$  consists of images. Predictions are obtained by model  $f : \mathcal{X} \rightarrow \mathcal{Y}$  that is trained by minimizing a loss function  $\ell : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}^+$ . A labeled set  $S_m = \{(x_i, y_i)\}_{i=1}^m \subseteq (\mathcal{X} \times \mathcal{Y})^m$  is sampled i.i.d. from  $P(X, Y)$  which is the distribution over  $\mathcal{X} \times \mathcal{Y}$ . The empirical risk of classifier  $f$  is given by  $\hat{r}(f|S_m) \triangleq \frac{1}{m} \sum_{i=1}^m \ell(f(x_i), y_i)$  [22, 12].

A selective model is the pair of the already defined prediction function  $f$  and a selection function  $g^* : \mathcal{X} \rightarrow \{0, 1\}$ . which performs the binary task of abstaining for  $f$ . In this work a sample shall be rejected when it is predicted as “positive” for a possible fault. To be in accordance with [22]  $g^*$  is inverted to  $g$ .

$$g(x) = 1 - g^*(x) \quad (1)$$

Therefore, an input  $x$  is rejected as follows.

$$(f, g)(x) \triangleq \begin{cases} f(x), & \text{if } g^*(x) = 0; \\ \text{reject}, & \text{if } g^*(x) = 1. \end{cases} \quad (2)$$

Selective prediction can be evaluated by coverage and risk. The empirical coverage that is the ratio of data which is kept is defined as

$$\hat{\phi}(g|S_m) \triangleq \frac{1}{m} \sum_{i=1}^m g(x_i). \quad (3)$$

The empirical risk is given by

$$\hat{r}(f, g|S_m) \triangleq \frac{\frac{1}{m} \sum_{i=1}^m \ell(f(x_i), y_i) g(x_i)}{\hat{\phi}(g|S_m)} \quad (4)$$

which will result in the relative error-rate on the covered data when the 0-1 loss function is applied.

**Loss Theory.** Loss functions are used to give a metric for the performance of a machine learning model. It is the basis of AI training since the gradient of those functions dictate the direction of change to the network in every training step. For classification tasks the cross-entropy loss is often used which is given by

$$L = - \sum_{i=1}^M y_i \log_e(p_i) \quad (5)$$

where L is the resulting Loss, i is the index of a class with M being the total number of present classes, y is the target value and p is the actual value of the i-th class [23]. When the target value of the correct class is 1 while all others are 0 like it is the case for one-hot-encoding this collapses into negative log-likelihood and is only dependent on the output value of the correct class  $p_c$ .

$$L = -\log_e(p_c) \quad (6)$$

From the highest possible  $p_c < 0.5$  in the case of a wrong prediction and the softmax activation function it directly follows that the lower bound  $L_f$  of a false predicted sample is

$$L_f > -\log_e(0.5) \approx 0.693. \quad (7)$$

Moreover, the upper bound of the loss  $L_c$  from a sample which is predicted correctly is given by

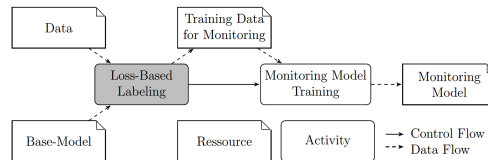
$$L_c < -\log_e\left(\frac{1}{n}\right) \quad (8)$$

when n classes are present. The upper and lower bound for false and correct predicted cases is infinity and zero, respectively. For  $n > 2$  there exists an overlap of samples that are correctly predicted with low confidence and samples that are incorrectly predicted. Discarding samples over a set loss threshold  $t < L_f$  evades the overlap.

## 4. Approach

The goal is to detect samples which are being classified incorrectly. It is desired that a pre-trained blackbox model

is monitored and distinction is possible for the whole range of reported output probabilities. The solution is to distinguish between samples of data that induce a high and low loss in the given base-model and further abstain from former cases.



**Figure 1:** Flow diagram of methodology where the gray box is the main contribution.

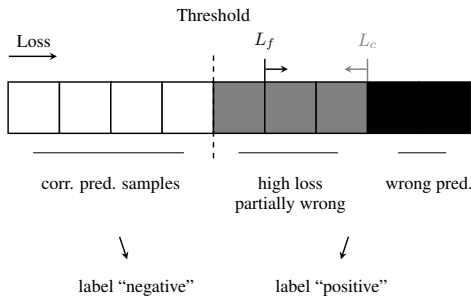
The proposed methodology expects an already trained base-model and additional data which is i.i.d. as the train and test data. The paradigm how the trained monitoring model is obtained is shown in Fig. 1. The loss-based labeling is depicted in Fig. 2. It is suitable to separate samples by a loss threshold  $t < L_f$  as derived from Eq. (7), leading to a division between correct and incorrect (over-confident) decisions. When a bigger ratio of samples leading to incorrect compared to correct predictions is rejected the safety of the AI algorithm is improved.

**Contribution.** The loss-based labeling provides a dataset where the weak points of the base-model are embedded and enables to intercept incorrect predictions for all output probabilities made by a blackbox model.

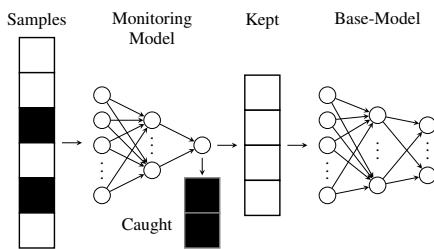
**Note.** The suggestion to use rejection is not our contribution; it was already proposed in the past [10, 18]. Our focus lies on intercepting over-confident errors and evaluating efficiency for the whole range of output probabilities.

The dataset consists of the unaltered base samples with replaced labels, corresponding to two classes “negative” and “positive”. The monitoring network is trained on mentioned dataset. Upon deployment the monitoring model will perform the binary decision prior to the base-model’s prediction as depicted in Fig. 3, reducing the error-rate.

The approach is especially helpful when considering that a trained classifier will have certain latent weak points. Uncovering those would be ideal but might be impossible for blackbox approaches. Even without exactly defining such weak points, the monitoring model can be able to learn a pattern which is present in critical input data. To best acquire the performance of the base-model by the monitoring model, data that the base-model has not seen during training is necessary. Since ANN training aims to reduce errors on the training set, weak points which the model includes may not be detectable at this stage. However, the monitoring model can extract further information on additional data.



**Figure 2:** Samples evaluated by base-model and sorted by increasing cross-entropy loss from left to right. The small black arrow gives the direction in which wrong predictions are located, the gray arrow shows the analog for correct predictions. The threshold is variable.



**Figure 3:** Ideal behaviour of monitoring model and base-model stack. Light and black boxes depict samples which will be predicted correct and incorrect, respectively.

One has to consider that the monitoring network does not specifically discriminate between correct and incorrect predictions but is rejecting upon the set loss threshold. Therefore, a performance metric for the monitoring network alone will not give sufficient information. The investigation includes which images are ultimately assigned to an incorrect class. The error-rate is solely reduced by rejecting such samples.

## 5. Evaluation and Experiments

### 5.1. Evaluation Goals

This work performs a study of the proposed methodology and compares this to the performance of a decision threshold based on the softmax output which is known to result in good results for selective prediction [22] but comes with the discussed shortcomings that the presented approach aims to solve. The objective is to answer the following research questions (RQs):

**RQ1.** Can rejected inputs be assigned to “weak points” of the base-model?

Answering this question will not explain exact features

on which the rejection is based. However, it helps to interpret whether the obtained monitoring is acting on a meaningful basis.

**RQ2.** Is the reduction of the error-rate by rejection based on the monitoring model better than pure chance?

Since even random rejection will improve the error-rate by a factor of the rejection rate, it is important to investigate whether the monitoring is resulting in an improvement higher than this base-line.

**RQ3.** Are incorrect decisions caught for the whole range of output probabilities?

This work is motivated on catching over-confident decisions, therefore, answering this question is a key aspect of the evaluation. Incorrect decisions with high output probabilities might be challenging but are the most important inputs to reject.

**RQ4.** What are the differences for both datasets?

It is important to point out where differences occur since this can indicate specific limitations of the method.

### 5.2. Datasets

To evaluate whether the approach can achieve improved safety of a model the GTSRB [24] is chosen. Since it is an automotive related dataset incorrect classification may lead to fatal decisions. For the sake of minimizing threats to validity the approach is evaluated on an additional dataset. Fashion-MNIST (F-MNIST) [25] is chosen for this purpose for multiple reasons. First, the classification is based on pictures of real world objects which is comparable to GTSRB. Secondly, there are less different classes which allows for a more detailed analysis. Moreover, although F-MNIST is relatively low dimensional, the SOTA error-rate of over 3% [26] is comparable high. Since [14] showed that detecting wrong predictions may work on simpler datasets but fail on more complex ones, for this work it was decided against classical MNIST [27]. The evaluation is only shown for those two datasets due to space limitations.

**German Traffic Sign Recognition Benchmark.** The GTSRB dataset is intended as a automotive related large multi-category classification benchmark. It consists of 39209 train and 12630 test samples corresponding to 43 different categories. Distribution of classes are highly unbalanced such that some classes are almost ten times as frequently present as other classes [24]. The images were extracted from video sequences and are supplied as RGB color images of different sizes between  $15 \times 15$  and  $222 \times 193$  pixels. In this work color channels are kept but normalized to one and images are resized to  $32 \times 32$  pixels. No action is performed that is tackling the unbalanced distribution of classes.

**Fashion-MNIST.** The F-MNIST dataset consists of gray scale images based on Zalando’s article images. There are 60,000 train and 10,000 test samples that are grouped

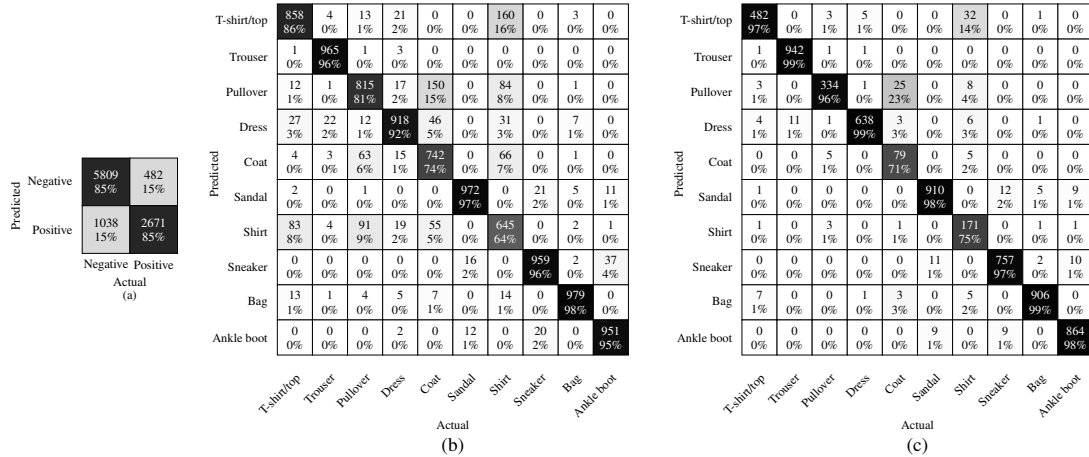


Figure 4: (a) Confusion matrix of a F-MNIST monitoring model with  $t = 0.1$  (b) confusion matrix of the F-MNIST base-model (c) base- combined with a monitoring model where  $t = 0.1$ .

into ten different classes. Each image has a dimension of  $28 \times 28$  pixels. In this work the images were preprocessed by normalizing the pixel values to one.

### 5.3. Experimental Setup

For the GTSRB dataset a convolutional neural network (CNN) with the LeNet-5 architecture [28] while for the F-MNIST dataset a fully connected (FC) feed-forward neural network is applied by using the TensorFlow library [29] as listed in Tab. 1.

Table 1  
Parameters of the networks used in the experiments.

Dataset	GTSRB	F-MNIST
<b>Base-Model</b>		
Architecture	LeNet-5	FC, 1 hidden layer, 128 neurons
N Parameters	85 k	100 k
Base Accuracy	90.48%	88.04%
<b>Monitoring</b>		
Architecture	LeNet-5	FC, 2 hidden layers, 32 neurons each
N Parameters	85 k	26 k
Training Samples	7,840	10,000
Portion of Training Data	20%	17%

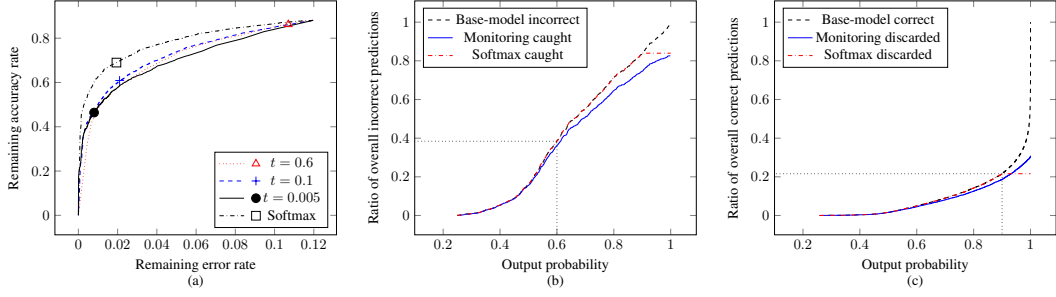
Ratios of training data for the monitoring model were kept to a minimum to have minor influence on the base-model training. However, a sufficient absolute number of images is needed to enable successful training convergence. To decide when to stop the monitoring training and to chose which loss threshold performs best, 10% of the training data used for the base-model is adapted as validation set. The result of the classification task performed by the monitoring model shows whether the data induces a higher loss than the given threshold on the base-model. Chosen threshold values are based on the

distribution of the loss for predictions by the base-model. The overall performance of the base- and monitoring-model stack was evaluated on the official test set which was not used in any of the training procedures. This is finally compared to the performance of the base-model on the same test data.

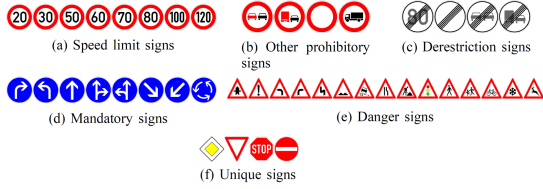
### 5.4. Results

**Loss-Based Monitoring.** The lowest loss for a sample that is incorrectly predicted is 0.699 for GTSRB and 0.696 for F-MNIST which places both near to the theoretical minimum given by Eq. (7). The maximum loss of correctly predicted samples is determined to be 1.757 for GTSRB and 1.353 for F-MNIST which is smaller than the given upper bound from Eq. (8). The confusion matrix of the trained monitoring model for F-MNIST is shown in Fig. 4 (a) in combination with confusion matrices of the base-model due to limited amount of classes. Data which is predicted “negative” is passed from the monitoring to the base-model for inference. Images that are flagged as “positive” are discarded due to them being considered unsafe inputs.

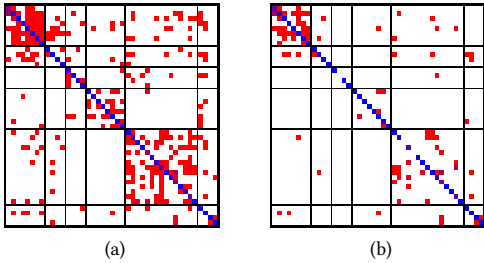
**Confusion Matrices (RQ1).** Figure 4 shows that for F-MNIST error-rates of all classes except the “coat” class improved. The classes “t-shirt/top”, “pullover”, “coat” and “shirt” are discarded the most. Additionally, those are the classes with the lowest accuracy in the base-model. Since there are 43 classes for the GTSRB no numbers are given but both base and monitored confusion matrix are color coded as shown in Fig. 7. Individual classes can be identified by Fig. 6. Comparing both matrices shows that less misclassification occurred. While three classes were fully rejected the error-rate was improved for all



**Figure 5:** (a) rar vs rer for F-MNIST for monitoring models with different loss thresholds  $t$  and rejection based on softmax where the operation points ( $d = 0.5$  for monitoring,  $d = 0.9$  for softmax) are marked. The operation points are aligned with Tab. 2. (b) Ratio of overall incorrect and caught predictions against output probabilities for monitoring with  $t = 0.1$  and softmax at marked operation point e.g. incorrect predictions with output probabilities of 60% or lower make up ca. 40% of all incorrect predictions. Almost all of those are caught by the monitoring (Zero gradient is bad). (c) Ratio of overall correct and discarded correct predictions against output probabilities for monitoring with  $t = 0.1$  and softmax e.g. correct predictions with output probabilities of 90% or lower make up ca. 20% of all correct predictions (Zero gradient is good).



**Figure 6:** All GTSRB classes grouped into subclasses such that classes of the confusion matrix in Fig. 7 can be identified, modified from [24].



**Figure 7:** (a) Base confusion matrix. (b) Monitored confusion matrix with  $t=0.005$ . Columns are actual while rows are predicted classes where values are given in relative color code, increasing from white (equal to zero) to red to blue (equal to 100% of the given class). Lines are separating the classes as grouped in Fig. 6 which does not match with the official numbering of classes. Best viewed in color. Illustration style adapted to own data from [24]

remaining classes except for one.

**Coverage and Error Trade-Off.** To evaluate the selective prediction it was decided against ROC-curves since the separation of “positive” and “negative” samples

does not reflect the effectiveness of the method. Instead, two metrics are adapted from [6]. The remaining error rate (rer) and remaining accuracy rate (rar) give the error-rate and correctly classified rate, respectively, relative to the overall input data including the rejected portion. The rer can be expressed with risk when calculated by the 0-1 loss and coverage via Eq. (3) and Eq. (4) as

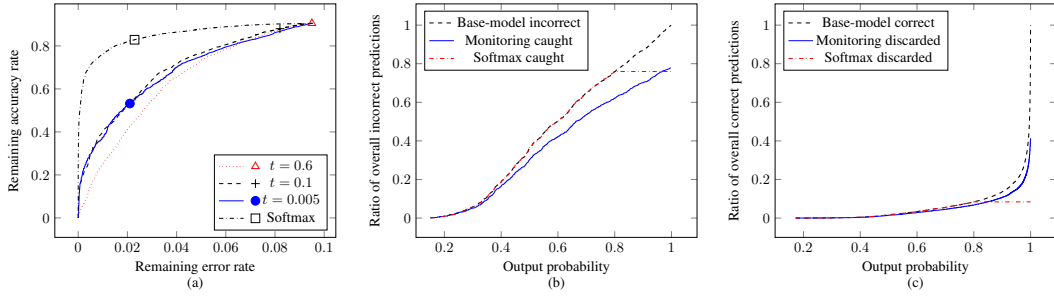
$$rer = \hat{r} \cdot \hat{\phi} \quad (9)$$

while the rar can be expressed as

$$rar = \hat{\phi} \cdot (1 - \hat{r}) = \hat{\phi} - rer. \quad (10)$$

**RQ2.** Figure 5 (a) and Fig. 8 (a) represent the rar against rer for both datasets and multiple monitoring networks where the decision boundary  $d$  of the binary classification was varied. Rejection based on softmax values are given as comparison. Each point on a function represents a possible operation point and corresponds to a different binary decision boundary value. Due to the counter-intuitive behaviour it has to be mentioned that while the decision boundary of the monitoring model is increased, the rer and rar increases since less samples are rejected. When the decision boundary is  $d \geq 1$ , all samples are kept which is analog to not applying the monitoring network resulting in rer being equal to the base error-rate and  $rar + rer = 1$ . However, for the softmax activation function it is vice versa, increasing the decision boundary rejects more samples with higher output probability, leading to a decrease in rer and rar.

**RQ3.** For the marked operation points in Fig. 5 (a) and Fig. 8 (a), Fig. 5 (b) and Fig. 8 (b) depict the gap between incorrect predictions and caught ones for all output probabilities. In contrast, Fig. 5 (c) and Fig. 8 (c) show the ratio of rejected but correct classifications compared to all correct predictions.



**Figure 8:** (a) rar vs rer for GTSRB for monitoring models with different loss thresholds  $t$  and rejection based on softmax where the operation points ( $d = 0.5$  for monitoring,  $d = 0.8$  for softmax) are marked. The operation points are aligned with Tab. 2. (b) Ratio of overall incorrect and caught predictions against output probabilities for monitoring with  $t = 0.005$  and softmax at marked operation point (Zero gradient is bad). (c) Ratio of overall correct and discarded correct predictions against output probabilities for monitoring with  $t = 0.005$  and softmax at marked operation point (Zero gradient is good).

**Table 2**

Results for the GTSRB and F-MNIST with  $t = 0.005$  and  $t = 0.1$  for the monitoring while  $d = 0.8$  and  $d = 0.9$  for rejection based on softmax, respectively. The binary decision boundary is  $d = 0.5$  for the monitoring models. All values given in %.

Dataset	GTSRB	F-MNIST
<b>Monitoring Model</b>		
rer (base-model)	2.11 (9.52)	2.08 (11.96)
rar (base-model)	53.21 (90.48)	60.83 (88.04)
Caught errors (rejection rate)	77.87 (44.69)	82.61 (37.09)
High probability errors caught	✓	✓
<b>Softmax Threshold</b>		
rer (base-model)	2.29 (9.52)	1.94 (11.96)
rar (base-model)	82.87 (90.48)	68.98 (88.04)
Caught errors (rejection rate)	75.95 (14.81)	83.77 (29.04)
High probability errors caught	X	X

**RQ4.** Table 2 summarizes the results where operation points of similar rer values are compared. Rejection occurred for the whole range of output probabilities when the monitoring-model was applied.

## 5.5. Discussion

**Loss-Based Monitoring.** The separation of samples is dependent on the set loss threshold  $t$ . Setting  $t$  just below  $L_f$  will guarantee to separate a maximum of correct from all incorrect predictions. While this is true in theory, it was determined that the monitoring-model does not properly learn a separation of both classes when the threshold is set near  $L_f$  as seen for the operation points in Fig. 5 (a) and Fig. 8 (a) for  $t = 0.6$ . In this work it was determined that the threshold needs to be set dependent on the loss distribution of the base model such that a sufficient amount of samples corresponding to the positive class is present.

**Confusion Matrices (RQ1).** Analyzing the confusion matrices for F-MNIST reveals that classes “coat” and “shirt” are discarded the most. By this the error-rate for “coat” is even increasing. While all other classes resulted

in a lower error-rate by rejecting less samples this suggests that it is hard to classify those classes correctly and the base-model fails there in a different, more fundamental, way.

For the GTSRB absolute values of the confusion matrices will not be discussed in detail since they cannot be interpreted from given Fig. 7 due to too many classes. However, it is clearly visible that in the monitored case less misclassification occur overall. Confusion between “speed limit signs” shows only little improvement while confusion between other subgroups is decreased. The classes that are rejected completely are one of a few with a relative class frequency of less than 1.0% [24]. Such problems could be eliminated by tackling the unbalanced distribution problem, however, for this work the approach shall be analyzed without heavy interventions. Why two classes in the “danger signs” subclass clearly increases in error-rate is unknown since the base-model showed relative good performance for those.

**RQ1.** For both datasets, classes with poor base-performance were rejected, which shows that the monitoring detects weak points. However, for GTSRB this led to slight deterioration of two individual classes that the base-model was able to classify with low error-rate.

**Coverage and Error Trade-Off (RQ4).** When analyzing the rar vs. rer trade-off it is visible that GTSRB is more challenging to discriminate between positive and negative samples than F-MNIST since the graph shows a faster increasing gradient while approaching lower rer. This can be explained by GTSRB consisting of higher dimensional images. Additionally, there are 43 classes where a high loss can be present for relatively clear separated, correct decisions, meaning predictions with a maximum probability far away from the second highest score.

**RQ2.** Overall one can state, that the monitoring is efficiently improving the safety as long as the relative

reduction of the rer is greater than the relative reduction of the rar. This is true for the investigated operation points. If it does not hold, the result of rejection is only as efficient as discarding samples by pure chance.

**RQ3.** The ratio of caught samples against output probability as depicted in Fig. 5 (b) and Fig. 8 (b) confirms that the goal to discard over-confident predictions is reached. The graphs are continuously increasing and show no area with zero gradient. This implies that cases with various output probabilities were caught. In contrast, rejection based on the softmax threshold inherently missed over-confident errors but did not discard any high probability correct decisions. The latter explains the lower rar trade-off.

**RQ4.** While for F-MNIST, the gradient of the ratio of correct samples grows much faster than for discarded samples, the GTSRB dataset revealed to have a bigger portion of correct samples rejected. This is in accordance to the harsher rar trade-off. Overall the monitoring is less prone to reject high confident correct cases.

## 5.6. Limitations

A key aspect in the training of the monitoring is that it shall be based on data, that the base-model was not trained on to uncover weak points. In this work latter is accomplished with a portion of the base training data. One can argue that the base-model would show higher performance when this missing data would be available during training. In the following this is discussed separately for both datasets. Comparing the error-rate of state of the art networks for F-MNIST given as 3.09% with 3.2 M parameters [26] which is orders of magnitude more than the model investigated here, proves that reducing the error rate on the base-model to such levels is not an easy task. Given the size of the used model, obtaining such improvements is excluded but can be obtained by applying the monitoring. For GTSRB other works [30] report a error-rate smaller than 1.0% using the Le-Net architecture. However, the focus is an enhanced architecture and pre-processing of data that tackles the unbalanced class distribution and image quality. A common method is data augmentation [31] to alter class distribution. The gained performance is expected to heavily rely on such pre-processing which was consciously excluded in this work to not rely on balanced classes or other modifications that may induce any bias.

## 6. Conclusion

We motivated the need to reduce the error-rate of a base-model by catching over-confident errors due to their safety critical nature. We contributed a loss-based labeling that reflects the weak points of the base-model and

proposed to train a monitoring on the generated dataset to improve the well-known rejection procedure. We applied this approach to the GTSRB and F-MNIST dataset and compared it to rejection based on the softmax activation function. The presented empirical results showed that the error-rate was improved and over-confident predictions were successfully caught. We discussed that while the rejection based on a softmax threshold shows a better remaining accuracy rate trade-off, the range of output probabilities for caught samples is bigger for the monitoring approach. The shown results serve as a proof-of-concept for the approach which is targeting safety critical domains. We believe that the methodology can be used for a variety of models and datasets.

## Acknowledgments

This work was funded by the German Federal Ministry of Education and Research (BMBF) within the KI-ASIC project (16ES0993). We thank Infineon Technologies AG for supporting this research.

## References

- [1] D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, D. Mané, Concrete problems in ai safety, arXiv preprint arXiv:1606.06565 (2016).
- [2] S. Burton, L. Gauerhof, C. Heinzemann, Making the case for safety of machine learning in highly automated driving, in: International Conference on Computer Safety, Reliability, and Security, Springer, 2017, pp. 5–16.
- [3] P. Koopman, M. Wagner, Autonomous vehicle safety: An interdisciplinary challenge, IEEE Intelligent Transportation Systems Magazine 9 (2017) 90–96.
- [4] R. Salay, R. Queiroz, K. Czarnecki, An analysis of iso 26262: Using machine learning safely in automotive software, arXiv preprint arXiv:1709.02435 (2017).
- [5] C. Gabreau, B. Pesquet-Popescu, F. Kaakai, B. Lefèvre, Artificial intelligence for future skies: On-going standardization activities to build the next certification/approval framework for airborne and ground aeronautic products., in: AISafety@IJCAI, 2021.
- [6] M. Henne, A. Schwaiger, K. Roscher, G. Weiss, Benchmarking uncertainty estimation methods for deep learning with safety-related metrics., in: SafeAI@ AAI, 2020, pp. 83–90.
- [7] Y. Gal, Z. Ghahramani, Dropout as a bayesian approximation: Representing model uncertainty in deep learning, in: international conference on machine learning, PMLR, 2016, pp. 1050–1059.



- [8] C. Guo, G. Pleiss, Y. Sun, K. Q. Weinberger, On calibration of modern neural networks, in: International Conference on Machine Learning, PMLR, 2017, pp. 1321–1330.
- [9] J. Nixon, M. W. Dusenberry, L. Zhang, G. Jerfel, D. Tran, Measuring calibration in deep learning., in: CVPR Workshops, volume 2, 2019.
- [10] C.-K. Chow, An optimum character recognition system using decision functions, IRE Transactions on Electronic Computers EC-6 (1957) 247–254.
- [11] P. L. Bartlett, M. H. Wegkamp, Classification with a reject option using a hinge loss., Journal of Machine Learning Research 9 (2008).
- [12] Y. Geifman, R. El-Yaniv, Selectivenet: A deep neural network with an integrated reject option, in: International Conference on Machine Learning, PMLR, 2019, pp. 2151–2159.
- [13] N. Manwani, K. Desai, S. Sasidharan, R. Sundararajan, Double ramp loss based option classifier, in: Pacific-Asia Conference on Knowledge Discovery and Data Mining, Springer, 2015, pp. 151–163.
- [14] H. Jiang, B. Kim, M. Y. Guan, M. Gupta, To trust or not to trust a classifier, arXiv preprint arXiv:1805.11783 (2018).
- [15] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey, ACM computing surveys (CSUR) 41 (2009) 1–58.
- [16] A. Meinke, M. Hein, Towards neural networks that provably know when they don't know, 2020. arXiv:1909.12180.
- [17] I. Kononenko, Bayesian neural networks, Biological Cybernetics 61 (1989) 361–370.
- [18] C. Chow, On optimum recognition error and reject tradeoff, IEEE Transactions on information theory 16 (1970) 41–46.
- [19] C. M. Santos-Pereira, A. M. Pires, On optimal reject rules and roc curves, Pattern recognition letters 26 (2005) 943–952.
- [20] L. P. Cordella, C. De Stefano, F. Tortorella, M. Vento, A method for improving classification reliability of multilayer perceptrons, IEEE Transactions on Neural Networks 6 (1995) 1140–1147.
- [21] C. De Stefano, C. Sansone, M. Vento, To reject or not to reject: that is the question-an answer in case of neural classifiers, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 30 (2000) 84–94.
- [22] Y. Geifman, R. El-Yaniv, Selective classification for deep neural networks, arXiv preprint arXiv:1705.08500 (2017).
- [23] C. M. Bishop, N. M. Nasrabadi, Pattern recognition and machine learning, volume 4, Springer, 2006.
- [24] J. Stallkamp, M. Schlipsing, J. Salmen, C. Igel, The german traffic sign recognition benchmark: a multi-class classification competition, in: The 2011 international joint conference on neural networks, IEEE, 2011, pp. 1453–1460.
- [25] H. Xiao, K. Rasul, R. Vollgraf, Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, arXiv preprint arXiv:1708.07747 (2017).
- [26] M. S. Tanveer, M. U. K. Khan, C.-M. Kyung, Fine-tuning darts for image classification, in: 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, 2021, pp. 4789–4796.
- [27] Y. LeCun, The mnist database of handwritten digits, <http://yann.lecun.com/exdb/mnist/> (1998).
- [28] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE 86 (1998) 2278–2324.
- [29] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al., Tensorflow: Large-scale machine learning on heterogeneous systems, 2015.
- [30] A. Zaibi, A. Ladgham, A. Sakly, A lightweight model for traffic sign classification based on enhanced lenet-5 network, Journal of Sensors 2021 (2021).
- [31] L. Perez, J. Wang, The effectiveness of data augmentation in image classification using deep learning, arXiv preprint arXiv:1712.04621 (2017).